US 20070033402A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2007/0033402 A1**

Williams et al. (43) **Pub. Date: Feb. 8, 2007**

(54) **SYSTEM AND METHOD FOR PRE-LOADING PERSONAL MEDIA DEVICE CONTENT**

(76) Inventors: **Robert John Williams**, San Francisco, CA (US); **Darryl Wood**, Tacoma, WA (US); **Jay Jeffrey McLeman**, Issaquah, WA (US); **Adam Bruce Cappio**, Seattle, WA (US); **Sudheer Tumuluru**, Issaquah, WA (US)

Correspondence Address:
FISH & RICHARDSON PC
P.O. BOX 1022
MINNEAPOLIS, MN 55440-1022 (US)

(21) Appl. No.: **11/499,996**

(22) Filed: **Aug. 7, 2006**

Related U.S. Application Data

(60) Provisional application No. 60/705,969, filed on Aug. 5, 2005.

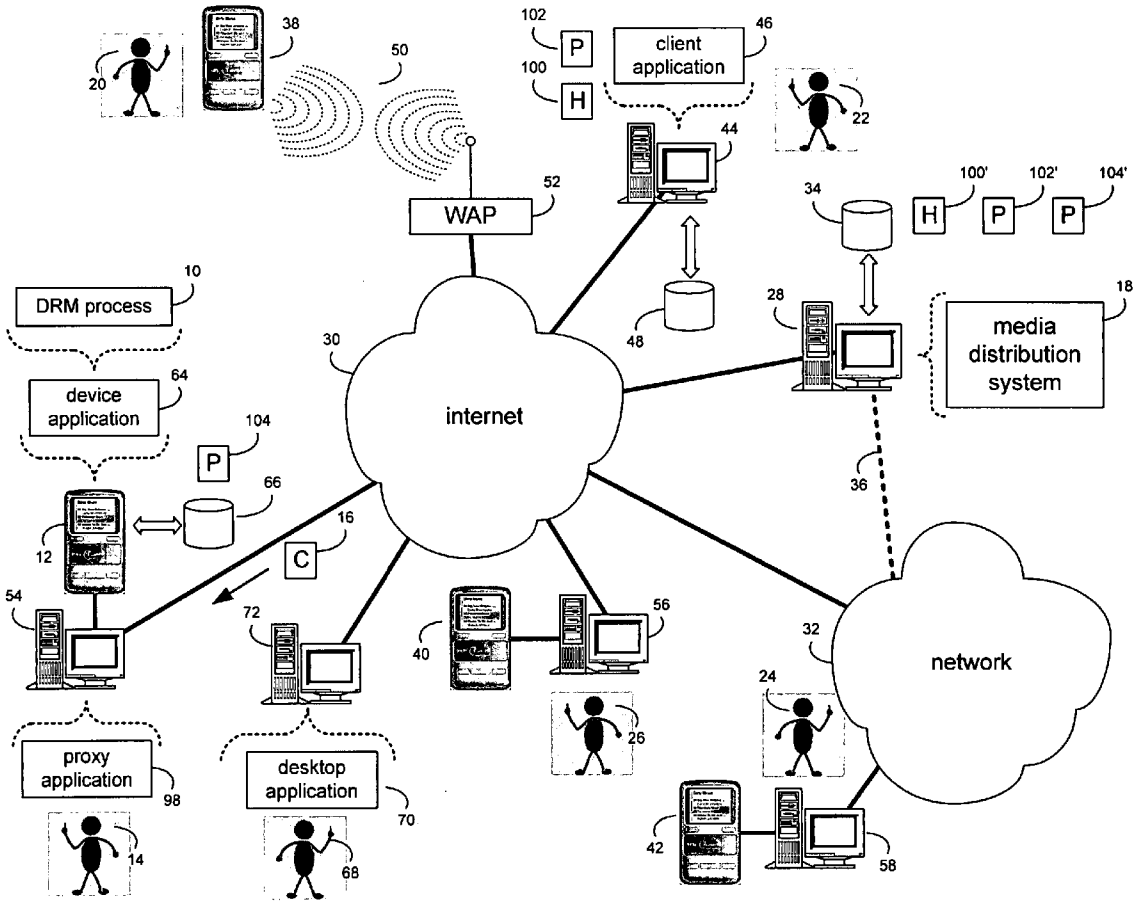
Publication Classification

(51) **Int. Cl.**
H04L 9/00 (2006.01)

(52) **U.S. Cl.** **713/169**

(57) **ABSTRACT**

A method, system and article for allowing access to pre-loaded media content on a personal media device. Access is allowed depending upon the presence of a pre-loaded content encryption key on the personal media device. The pre-loaded encryption key is associated with the pre-loaded media content.



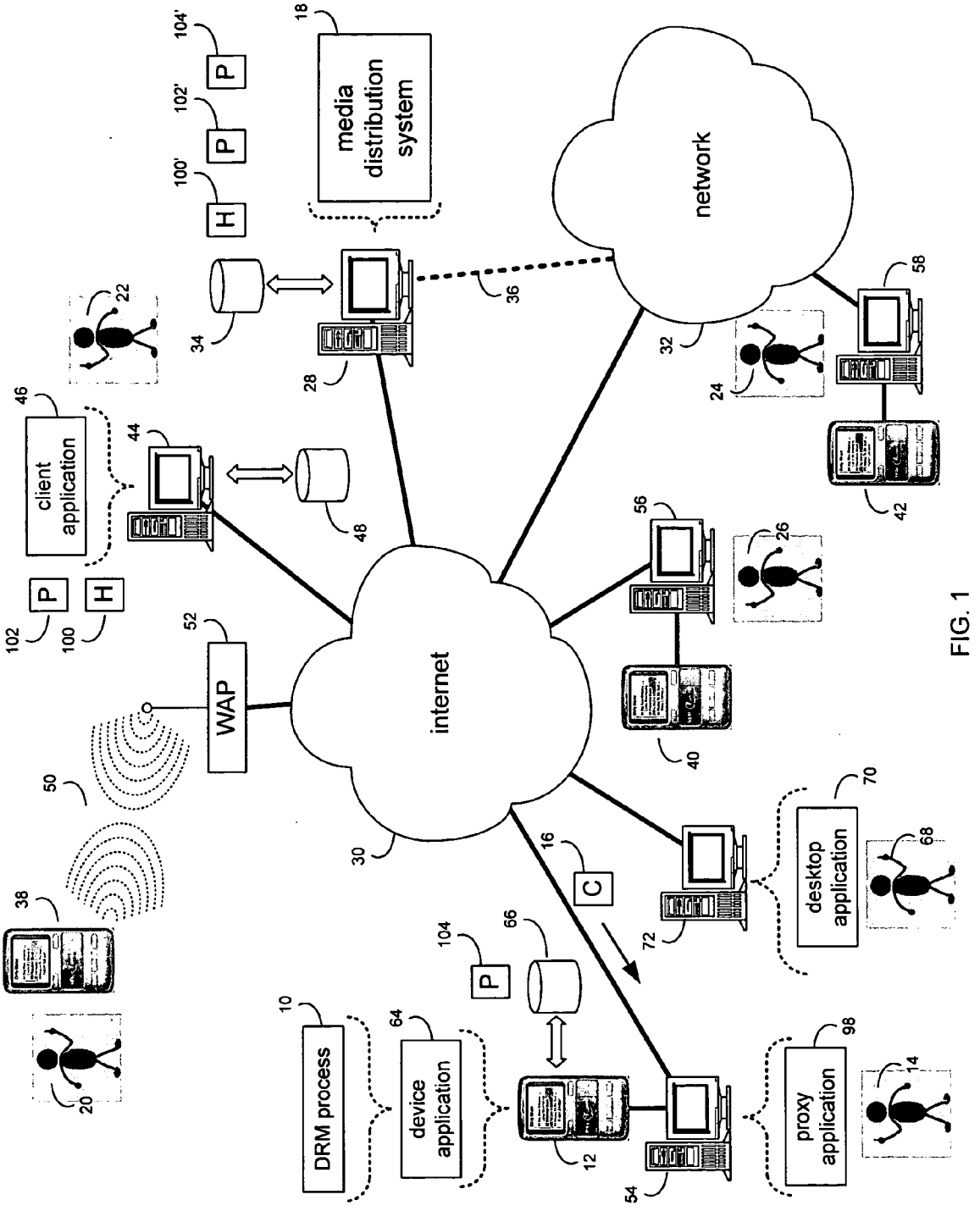

media distribution system
network
internet
WAP
client application
desktop application
DRM process
device application
proxy application
P
H
C

FIG. 1

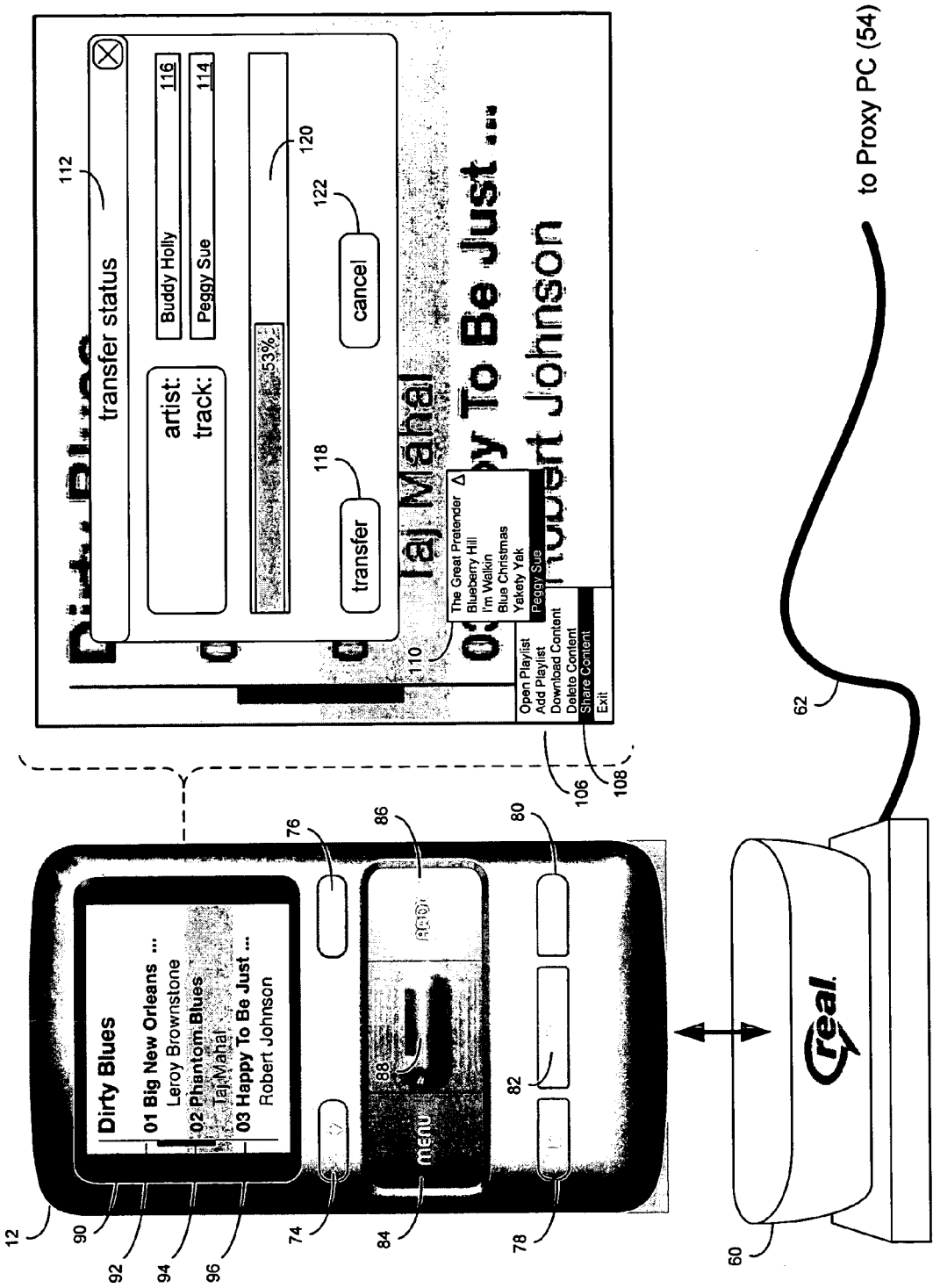

transfer status
artist:
track:
Buddy Holly
Peggy Sue
116
114
120
122
cancel
118
transfer
112
110
Taj Mahal
Robert Johnson
The Great Pretender
Blueberry Hill
I'm Walkin
Blue Christmas
Yakety Yak
Peggy Sue
Open Playlist
Add Playlist
Download Content
Delete Content
Share Content
Exit
106
108
Dirty Blues
01 Big New Orleans ...
Leroy Brownstone
02 Phantom Blues
Taj Mahal
03 Happy To Be Just ...
Robert Johnson
menu
12
90
92
94
96
74
76
84
86
78
80
82
88
60
62
to Proxy PC (54)

FIG. 2

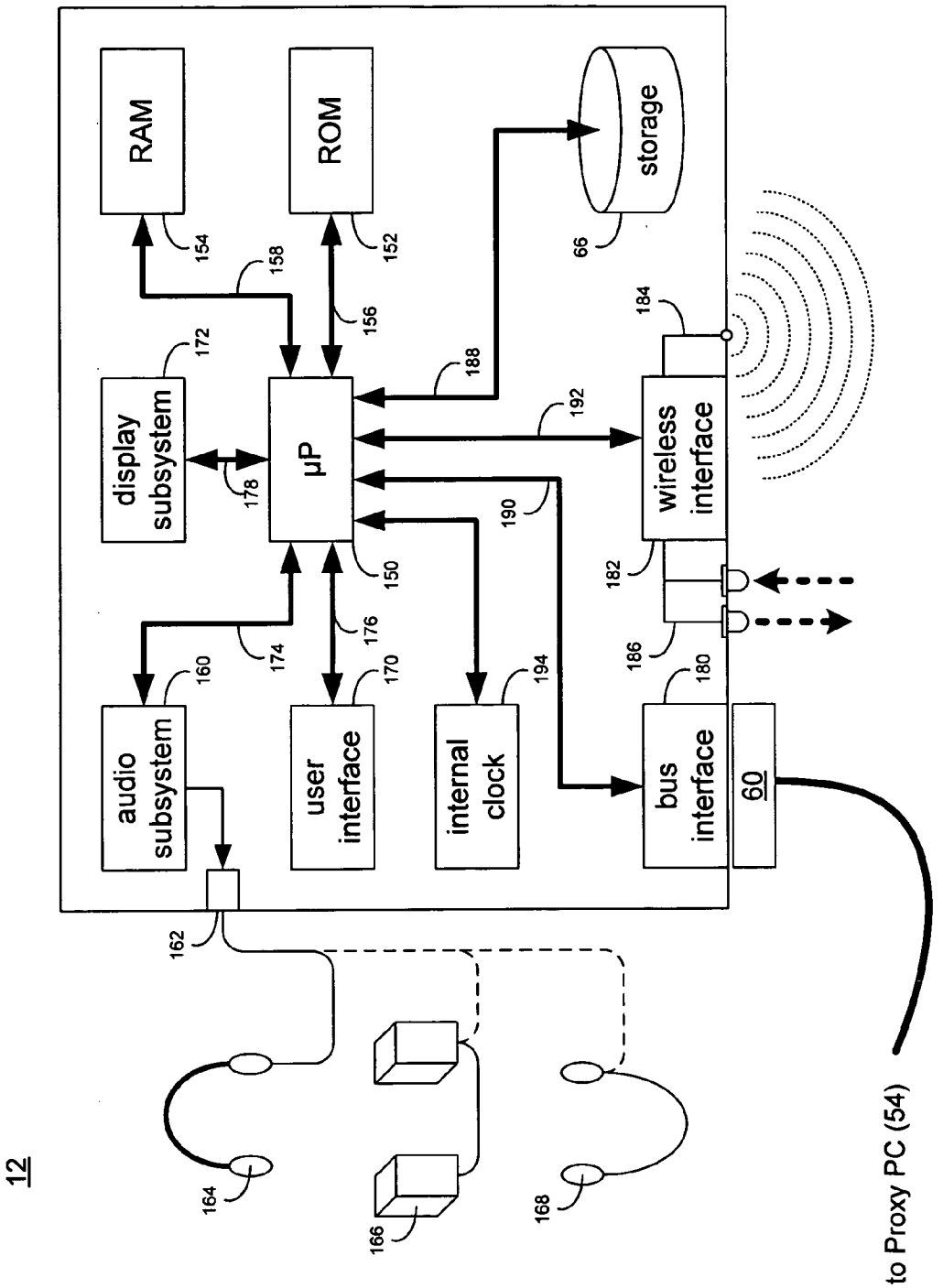

12
RAM
ROM
storage
display subsystem
µP
wireless interface
audio subsystem
user interface
internal clock
bus interface
60
to Proxy PC (54)
154
158
152
156
66
184
172
188
192
178
190
150
182
176
174
170
160
194
186
180
162
164
166
168

FIG. 3

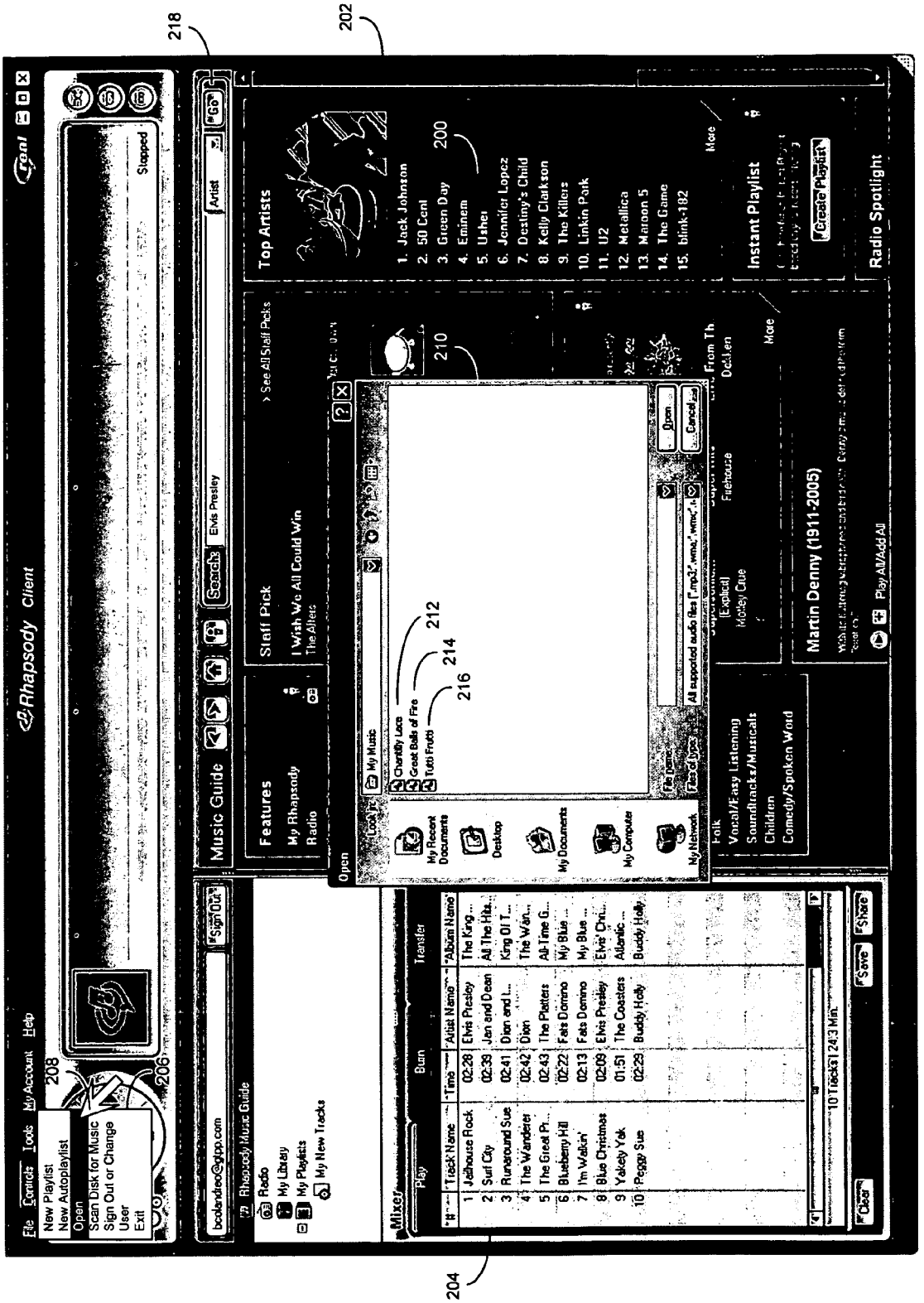

218
202
200
210
212
214
216
204
206
208
Rhapsody Client
Music Guide
Search
Elvis Presley
Features
Staff Pick
Top Artists
Radio Spotlight
Instant Playlist
Martin Denny (1911-2005)
Open
My Music
Chantilly Lace
Great Balls of Fire
Tutti Frutti
My Recent Documents
Desktop
My Documents
My Computer
My Network
Mixer
Play
Burn
Transfer
Track Name
Time
Artist Name
Album Name
Jailhouse Rock
Surf City
Runaround Sue
The Wanderer
The Great Pr...
Blueberry Hill
I'm Walkin'
Blue Christmas
Yakety Yak
Peggy Sue
Elvis Presley
Jan and Dean
Dion and t...
Dion
The Platters
Fats Domino
Fats Domino
Elvis Presley
The Coasters
Buddy Holly
The King ...
All The Hits ...
King Of T...
The Wan...
All-Time G...
My Blue ...
My Blue ...
Elvis' Chri...
Atlantic ...
Buddy Holly
Save
Share
Play All/Add All
Create Playlist
1. Jack Johnson
2. 50 Cent
3. Green Day
4. Eminem
5. Usher
6. Jennifer Lopez
7. Destiny's Child
8. Kelly Clarkson
9. The Killers
10. Linkin Park
11. U2
12. Metallica
13. Maroon 5
14. The Game
15. blink-182
More
Vocal/Easy Listening
Soundtracks/Musicals
Children
Comedy/Spoken Word
Folk

FIG. 4

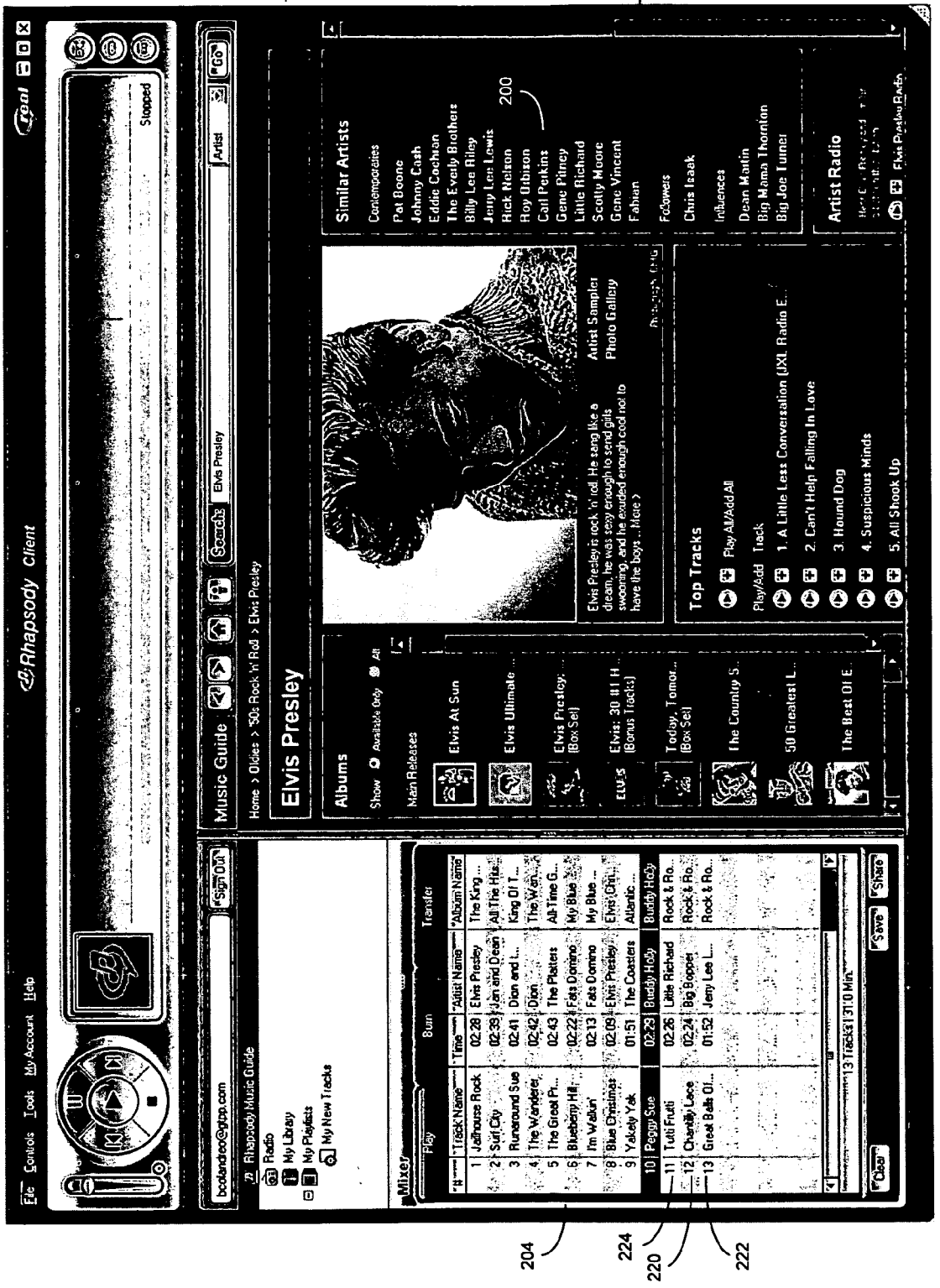

202
200
204
224
220
222
Rhapsody Client
Music Guide
Elvis Presley
Albums
Similar Artists
Top Tracks
Artist Radio
Mixer
Play
Burn
Transfer
Track Name
Time
Artist Name
Album Name
Clear
Save
Share

FIG. 5

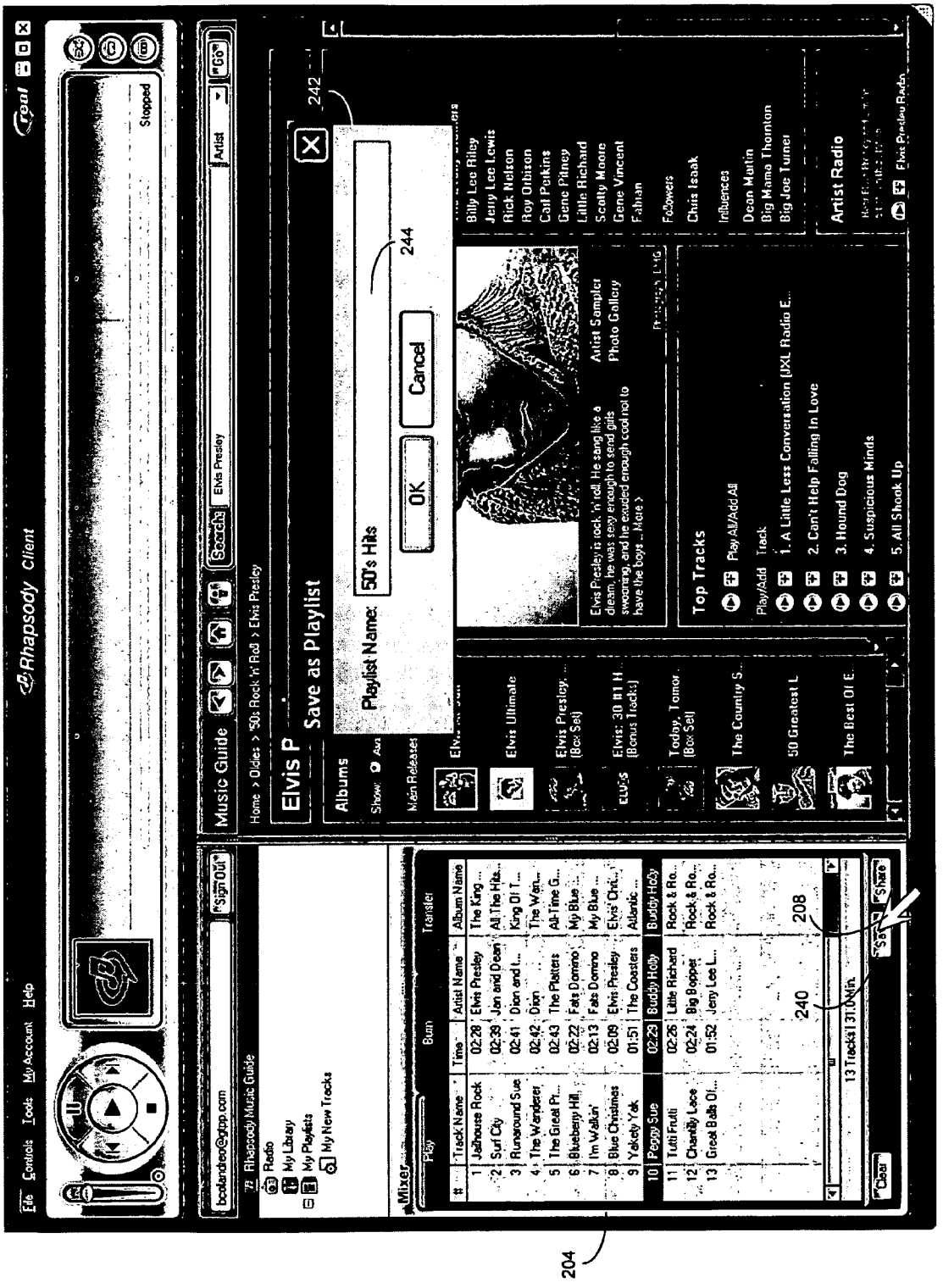

Rhapsody Client
Save as Playlist
Playlist Name:
50's Hits
OK
Cancel
Music Guide
Elvis Presley
Top Tracks
Artist Radio
204
208
240
242
244

FIG. 6

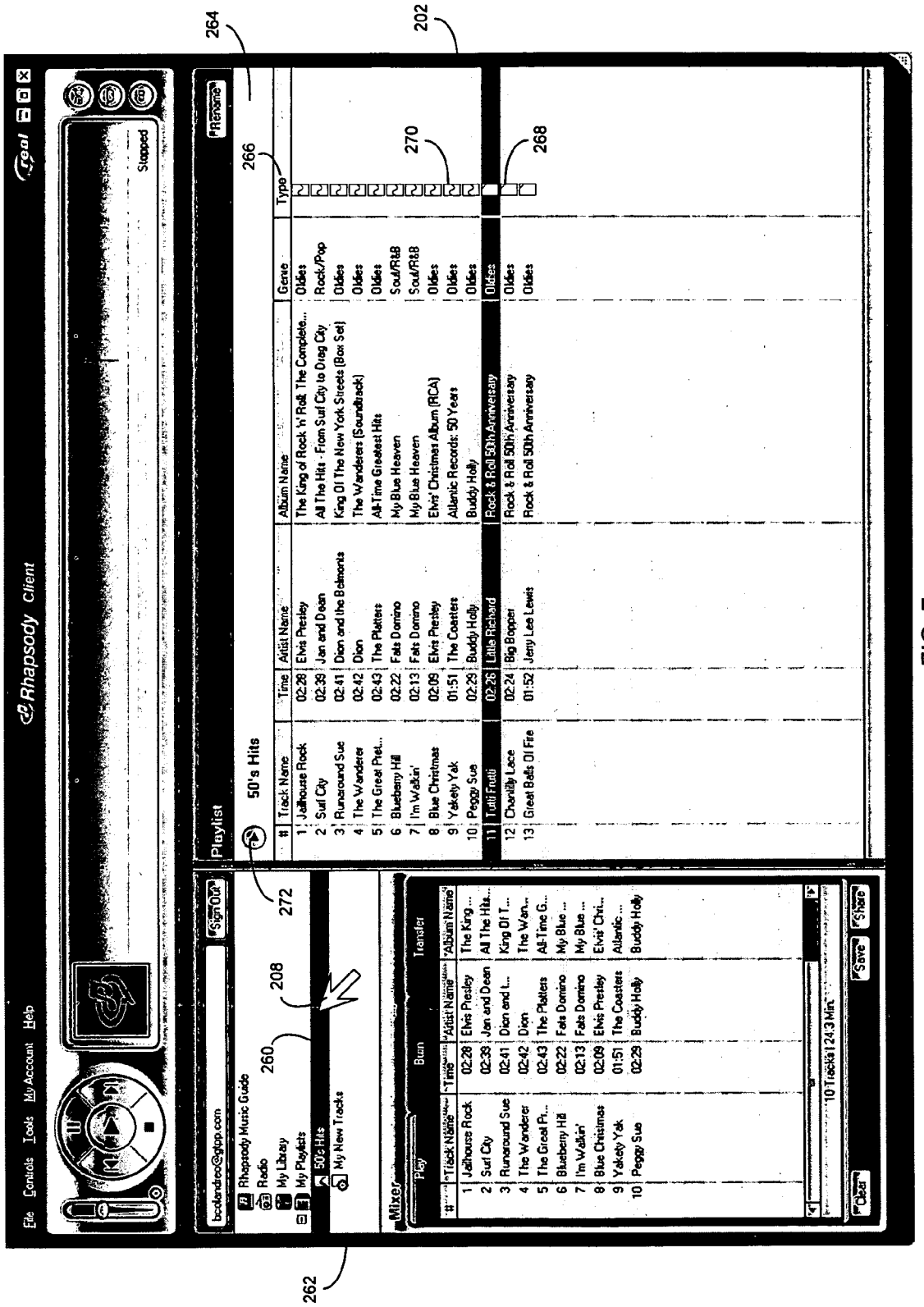

Rhapsody Client
File Controls Tools My Account Help
Stopped
bcdandeo@gtpp.com
Sign Out
Rhapsody Music Guide
Radio
My Library
My Playlists
50's Hits
My New Tracks
Mixer
Play
Burn
Transfer
Playlist
Rename
50's Hits
Track Name
Time
Artist Name
Album Name
Genre
Type
Clear
Save
Share
10 Tracks | 24.3 Min
262
260
208
272
264
266
202
270
268

FIG. 7

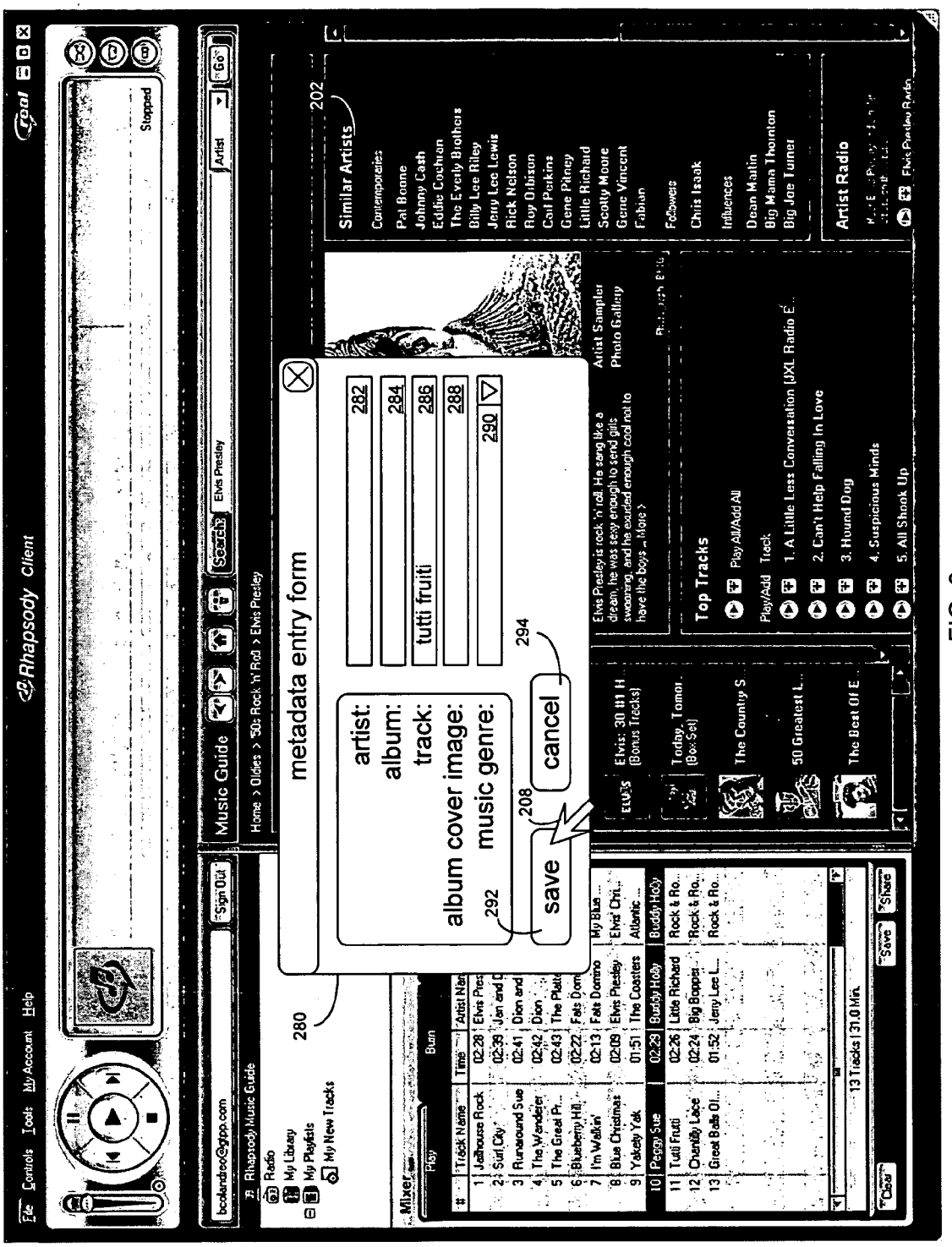

Rhapsody Client
metadata entry form
artist:
album:
track:
album cover image:
music genre:
tutti fruiti
save
cancel
280
282
284
286
288
290
292
294
208
202
Similar Artists
Top Tracks
Artist Radio

FIG. 8

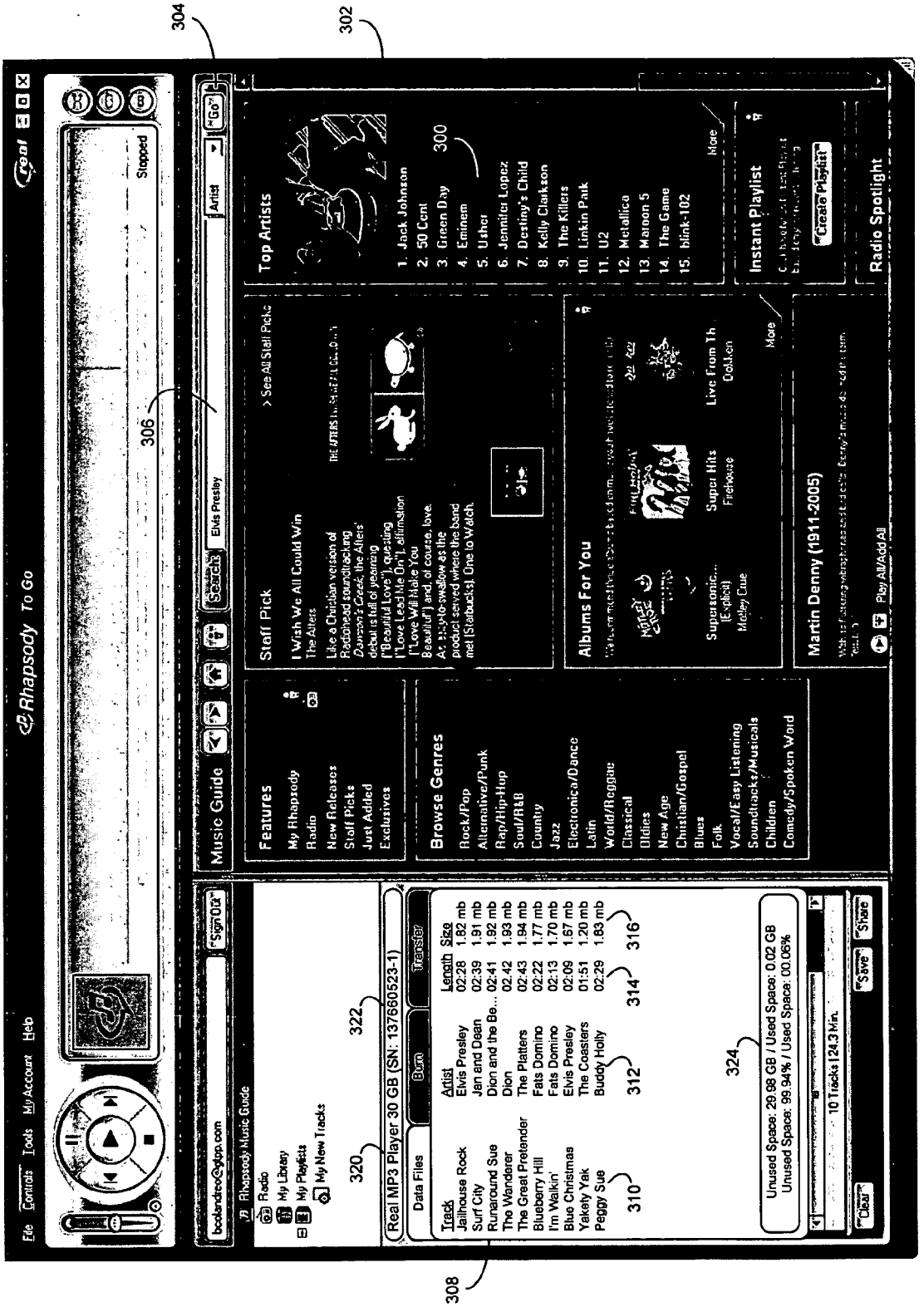

304
302
306
300
308
320
322
310
312
314
316
324
Rhapsody To Go
Music Guide
Features
Staff Pick
Top Artists
Browse Genres
Albums For You
Instant Playlist
Radio Spotlight
Martin Denny (1911-2005)
Real MP3 Player 30 GB (SN: 137660523-1)
Data Files
Burn
Transfer
Unused Space: 29.98 GB / Used Space: 0.02 GB
Unused Space: 99.94% / Used Space: 00.06%

FIG. 9

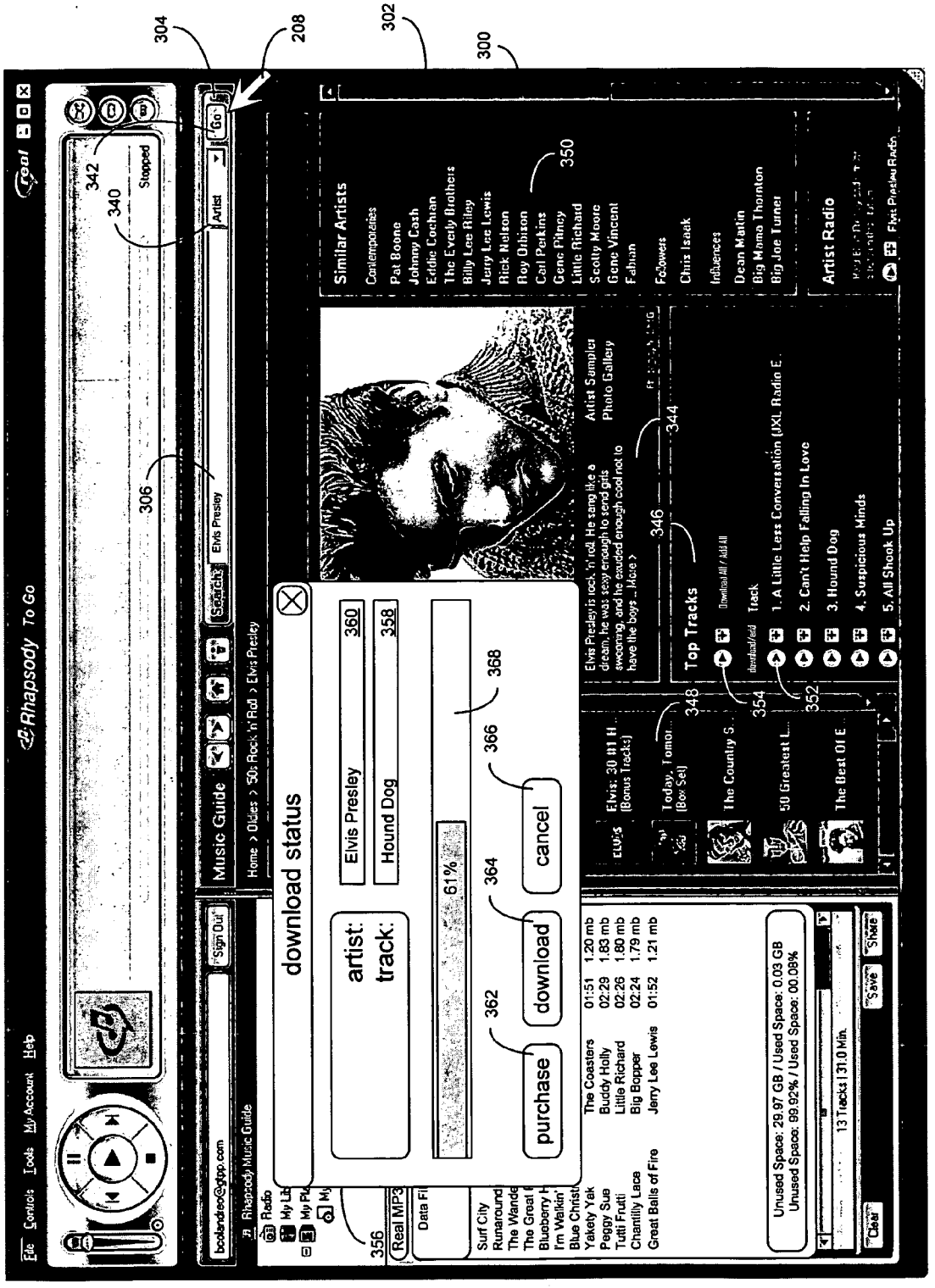

304
208
302
300
342
340
306
356
360
358
362
364
366
368
350
344
346
348
354
352
download status
artist:
track:
Elvis Presley
Hound Dog
61%
purchase
download
cancel
Similar Artists
Top Tracks
Artist Radio

FIG. 10

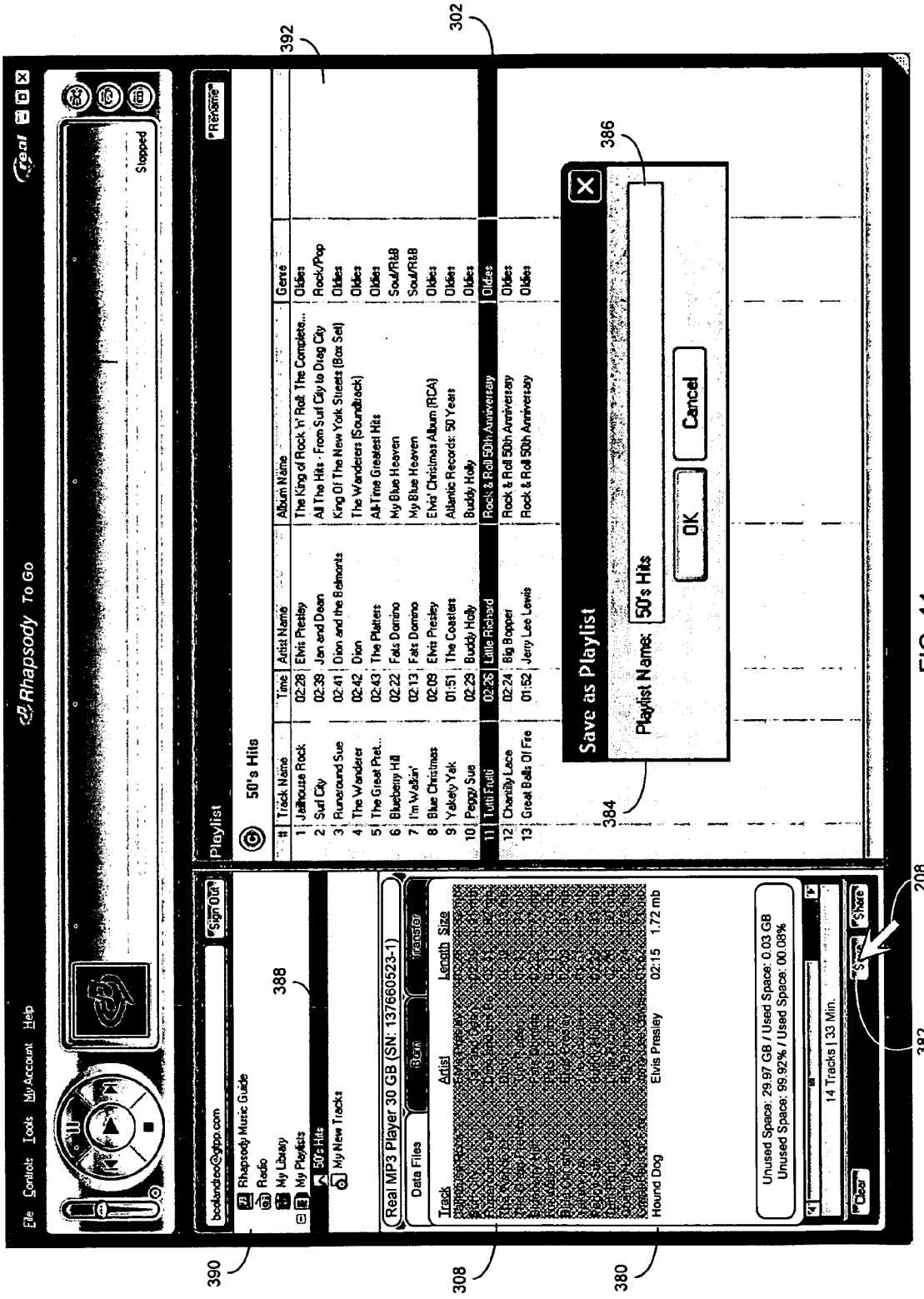

Rhapsody To Go
File Controls Tools My Account Help
Stopped
Sign Out
Rhapsody Music Guide
Radio
My Library
My Playlists
50's Hits
My New Tracks
388
390
Playlist
50's Hits
Rename
392
302
Track Name
Time
Artist Name
Album Name
Genre
1 Jailhouse Rock 02:28 Elvis Presley The King of Rock 'n' Roll: The Complete... Oldies
2 Surf City 02:39 Jan and Dean All The Hits - From Surf City to Drag City Rock/Pop
3 Runaround Sue 02:41 Dion and the Belmonts King Of The New York Streets (Box Set) Oldies
4 The Wanderer 02:42 Dion The Wanderers (Soundtrack) Oldies
5 The Great Pret... 02:43 The Platters All-Time Greatest Hits Oldies
6 Blueberry Hill 02:22 Fats Domino My Blue Heaven Soul/R&B
7 I'm Walkin' 02:13 Fats Domino My Blue Heaven Soul/R&B
8 Blue Christmas 02:09 Elvis Presley Elvis' Christmas Album (RCA) Oldies
9 Yakety Yak 01:51 The Coasters Atlantic Records: 50 Years Oldies
10 Peggy Sue 02:29 Buddy Holly Buddy Holly Oldies
11 Tutti Frutti 02:26 Little Richard Rock & Roll 50th Anniversary Oldies
12 Chantilly Lace 02:24 Big Bopper Rock & Roll 50th Anniversary Oldies
13 Great Balls Of Fire 01:52 Jerry Lee Lewis Rock & Roll 50th Anniversary Oldies
Save as Playlist
384
Playlist Name: 50's Hits
386
OK
Cancel
Real MP3 Player 30 GB (SN: 137660523-1)
308
Data Files
Music
Transfer
Track
Artist
Length
Size
Hound Dog
Elvis Presley
02:15
1.72 mb
380
Unused Space: 29.97 GB / Used Space: 0.03 GB
Unused Space: 99.92% / Used Space: 00.08%
14 Tracks | 33 Min.
Clear
Sync
Share
382
208

FIG. 11

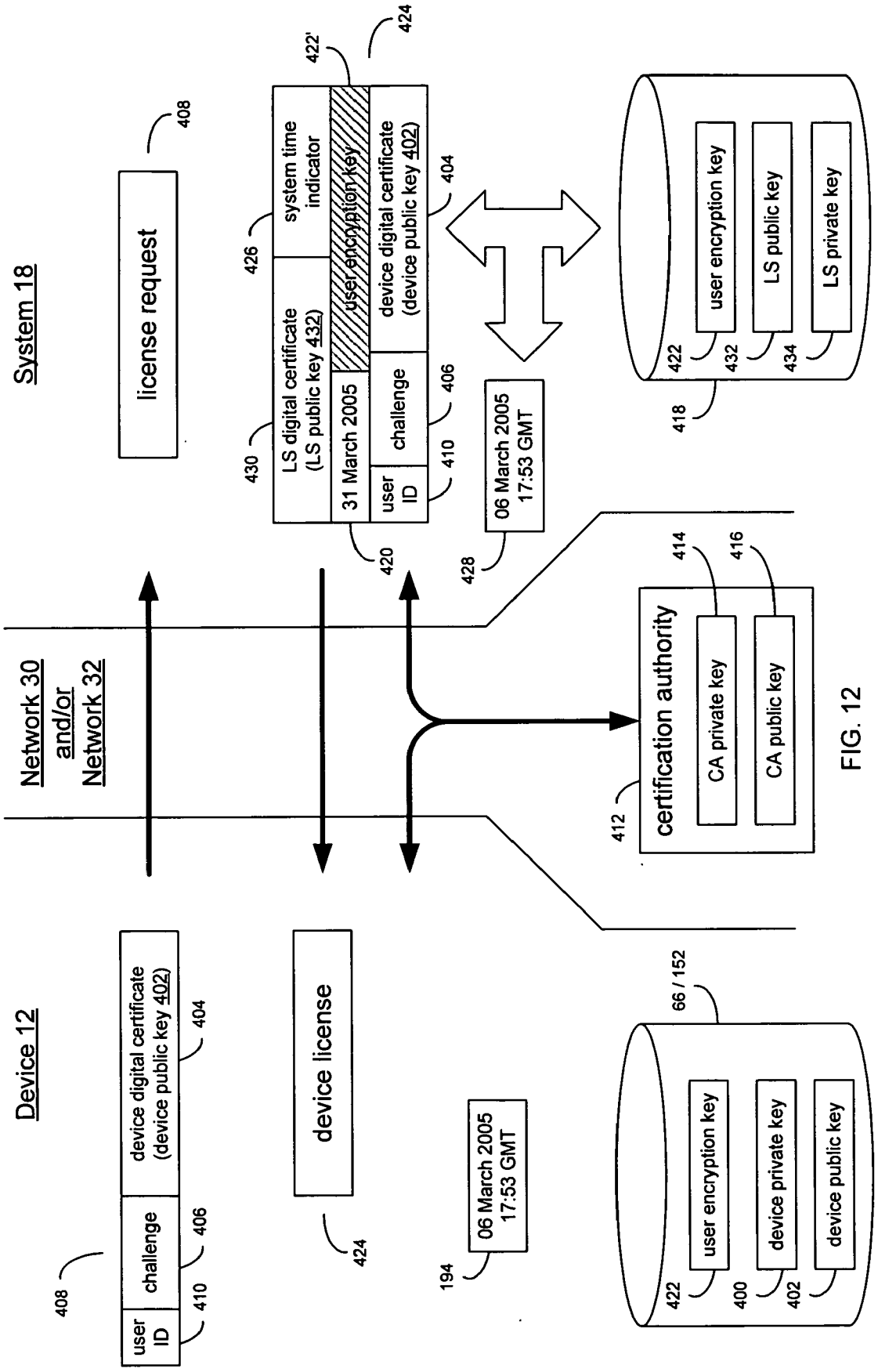

System 18
license request
408
LS digital certificate (LS public key 432)
system time indicator
user encryption key
device digital certificate (device public key 402)
422'
424
426
430
404
31 March 2005
challenge
406
user ID
410
420
06 March 2005 17:53 GMT
428
user encryption key
LS public key
LS private key
422
432
434
418
Network 30 and/or Network 32
certification authority
CA private key
CA public key
414
416
412
Device 12
device digital certificate (device public key 402)
404
challenge
406
user ID
410
408
device license
424
06 March 2005 17:53 GMT
194
66 / 152
user encryption key
device private key
device public key
422
400
402

FIG. 12

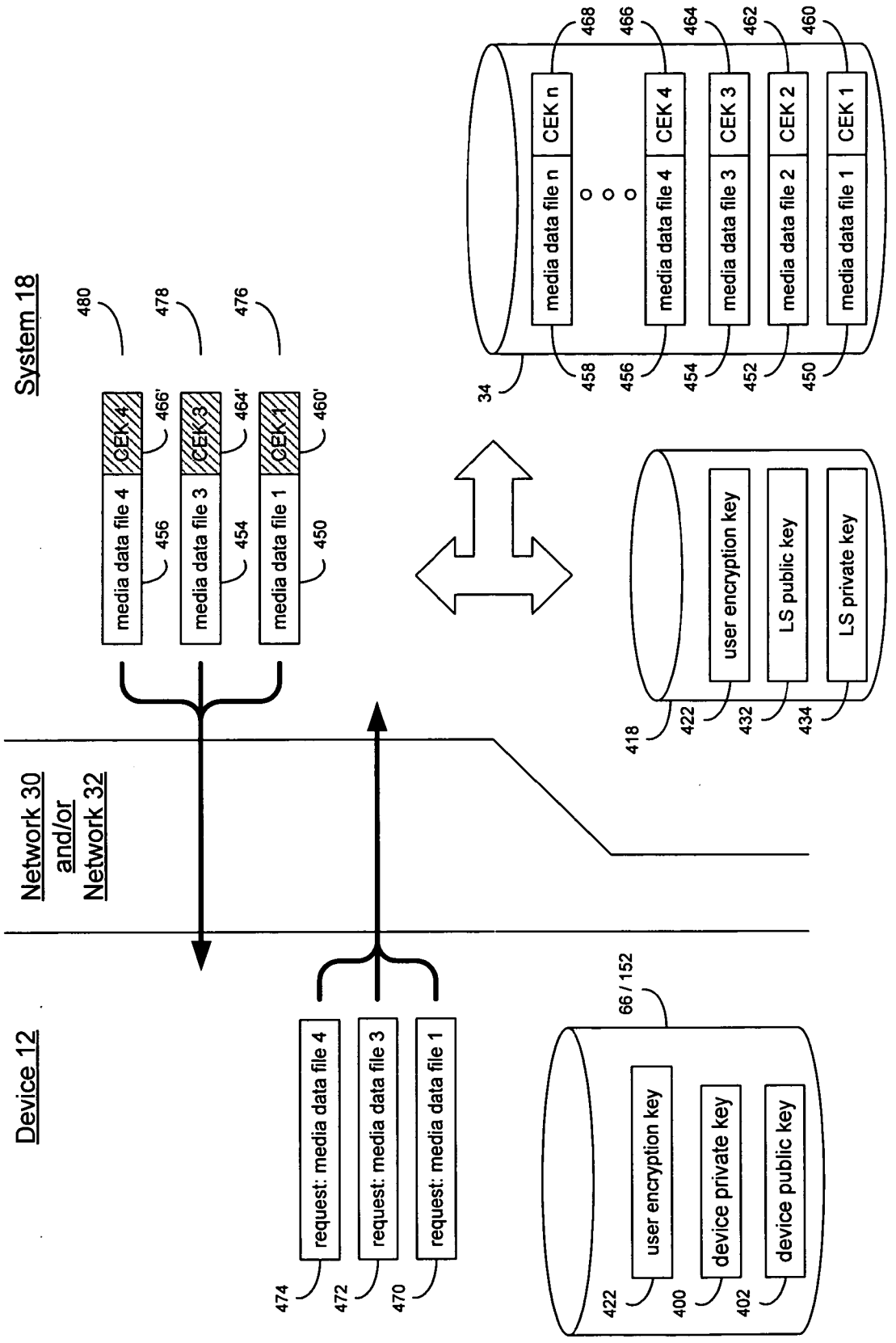

System 18
Network 30 and/or Network 32
Device 12
media data file 4
media data file 3
media data file 1
CEK 4
CEK 3
CEK 1
media data file n
CEK n
user encryption key
LS public key
LS private key
request: media data file 4
request: media data file 3
request: media data file 1
device private key
device public key
66 / 152
34
418
422
432
434
400
402
450
452
454
456
458
460
462
464
466
468
470
472
474
476
478
480

FIG. 13

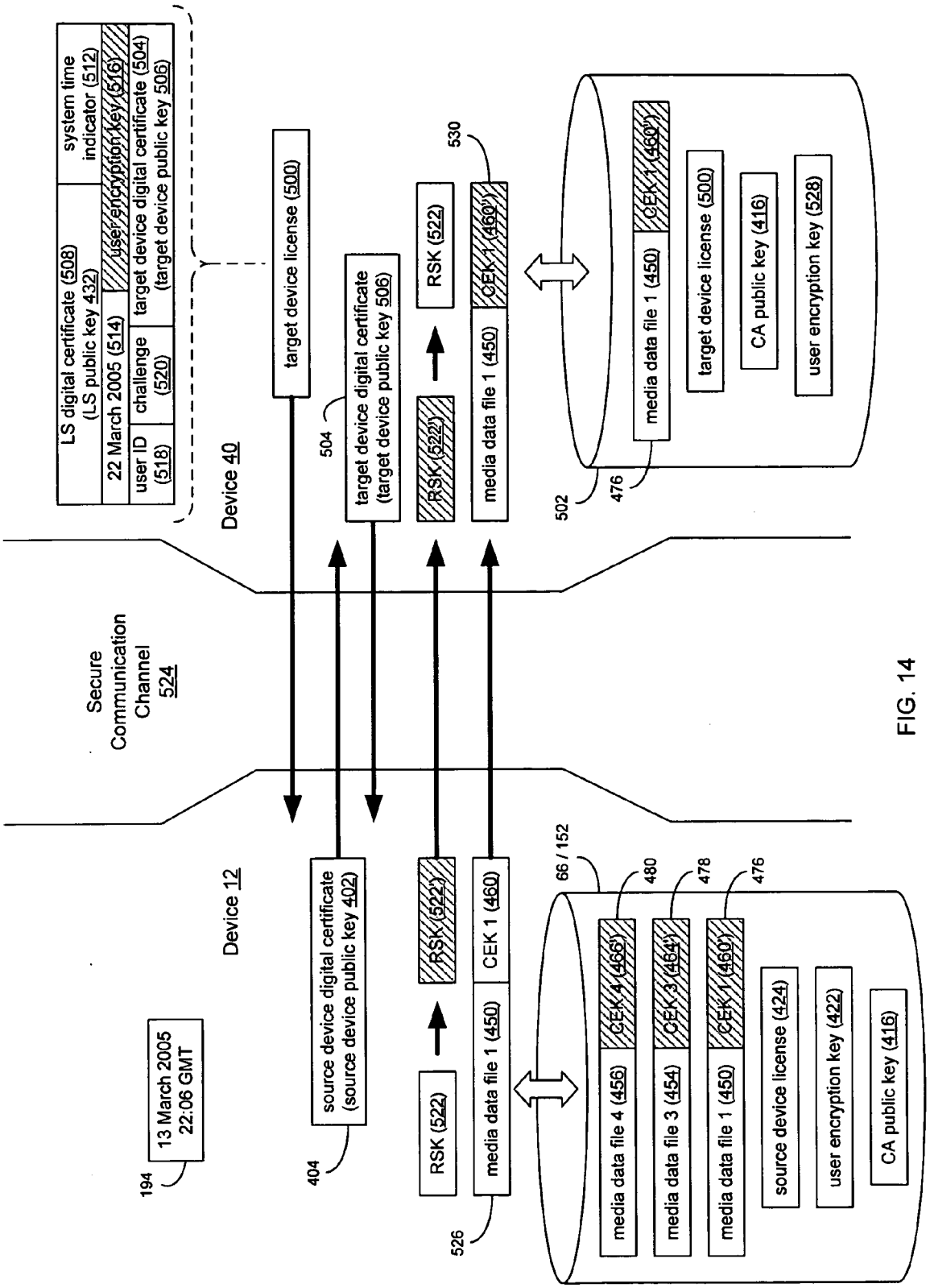

LS digital certificate (508)
(LS public key 432)
system time indicator (512)
22 March 2005 (514)
user encryption key (516)
user ID (518)
challenge (520)
target device digital certificate (504)
(target device public key 506)
Device 40
target device license (500)
504
target device digital certificate (target device public key 506)
RSK (522)
RSK (522)
CEK 1 (460)
530
media data file 1 (450)
502
476
media data file 1 (450)
CEK 1 (460)
target device license (500)
CA public key (416)
user encryption key (528)
Secure Communication Channel 524
Device 12
source device digital certificate (source device public key 402)
RSK (522)
RSK (522)
CEK 1 (460)
media data file 1 (450)
526
404
194
13 March 2005 22:06 GMT
66 / 152
480
478
476
media data file 4 (456)
CEK 4 (466)
media data file 3 (454)
CEK 3 (464)
media data file 1 (450)
CEK 1 (460)
source device license (424)
user encryption key (422)
CA public key (416)

FIG. 14

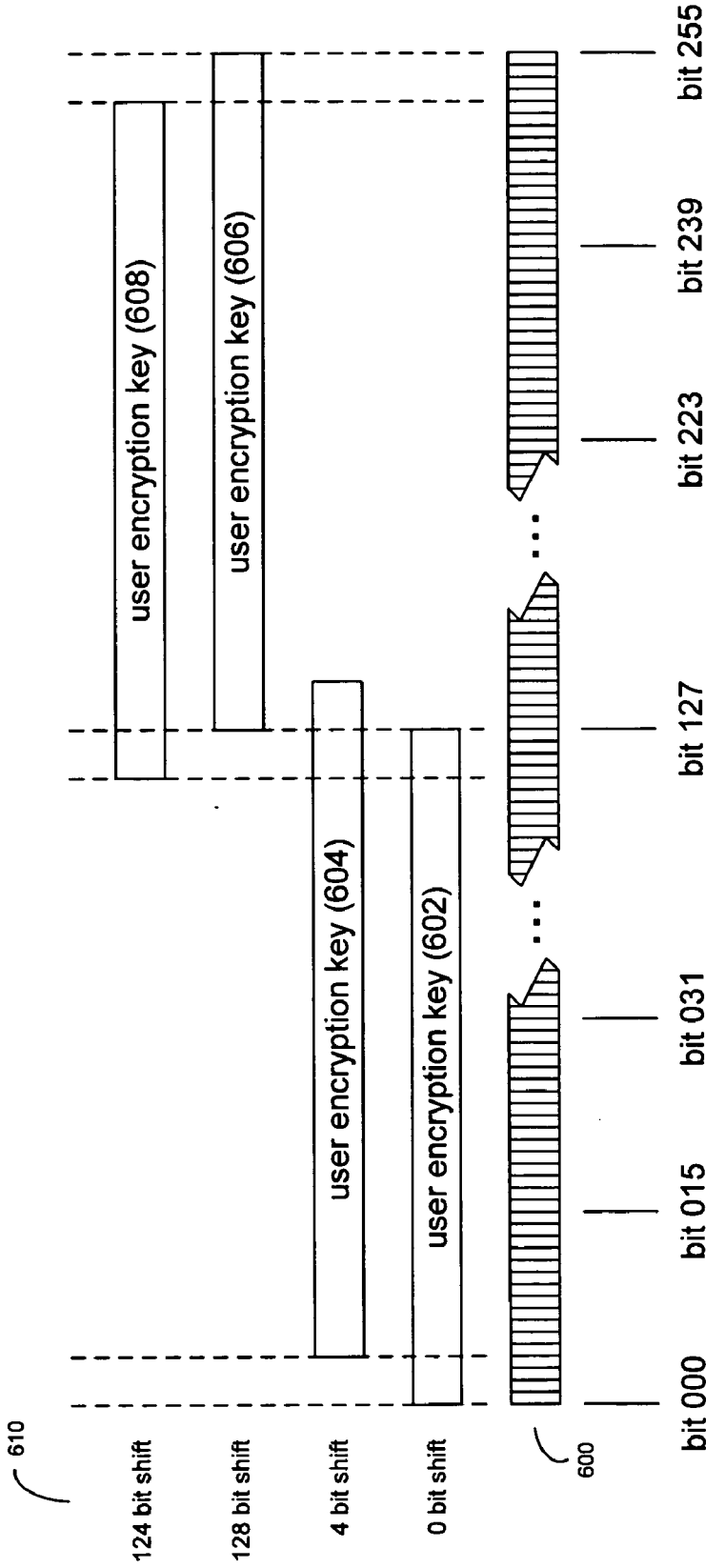

610
124 bit shift
128 bit shift
4 bit shift
0 bit shift
user encryption key (608)
user encryption key (606)
user encryption key (604)
user encryption key (602)
600
bit 000
bit 015
bit 031
bit 127
bit 223
bit 239
bit 255

FIG. 15

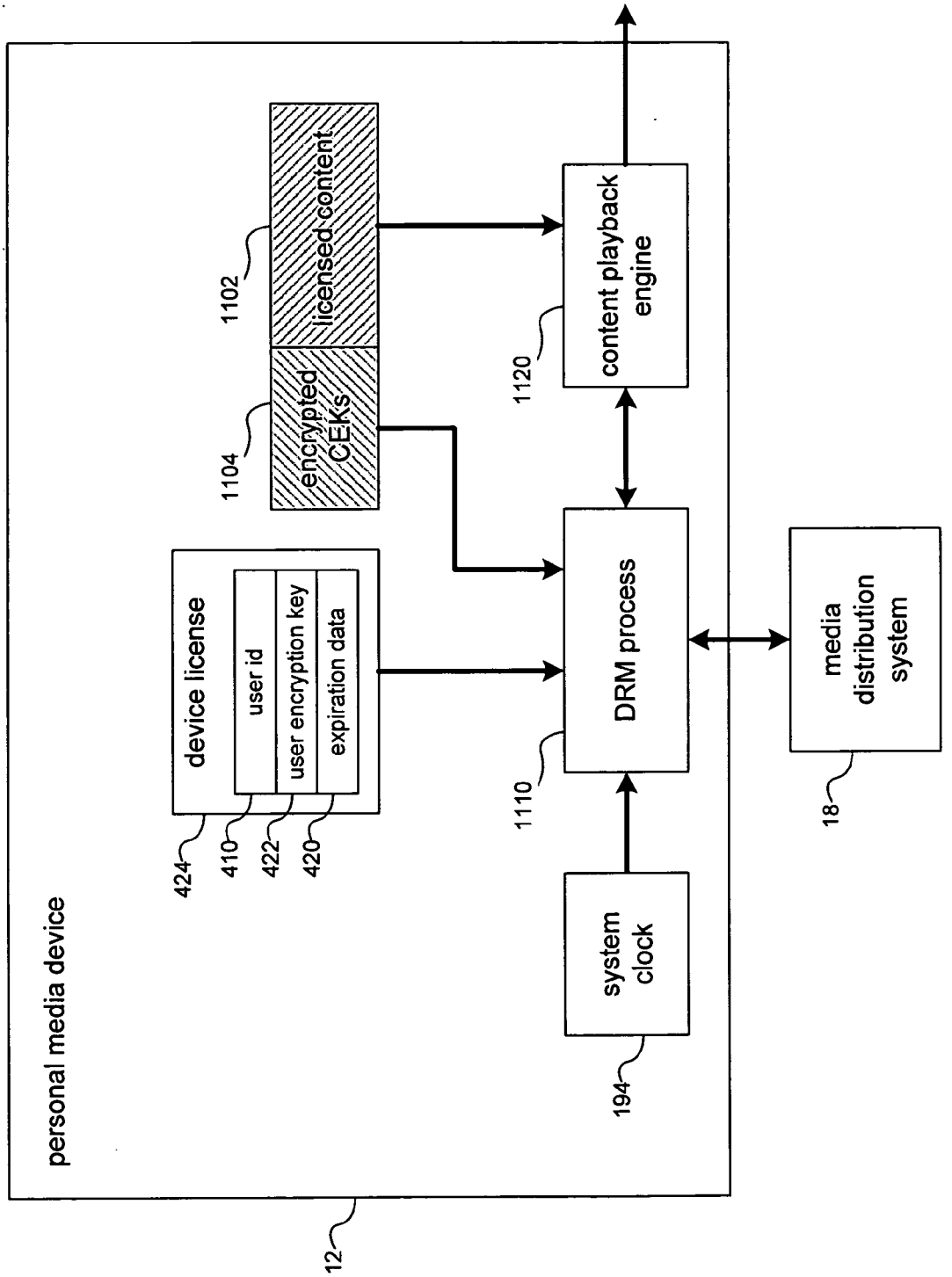

personal media device
12
licensed content
1102
encrypted CEKs
1104
content playback engine
1120
DRM process
1110
media distribution system
18
system clock
194
device license
user id
user encryption key
expiration data
424
410
422
420

FIG. 16

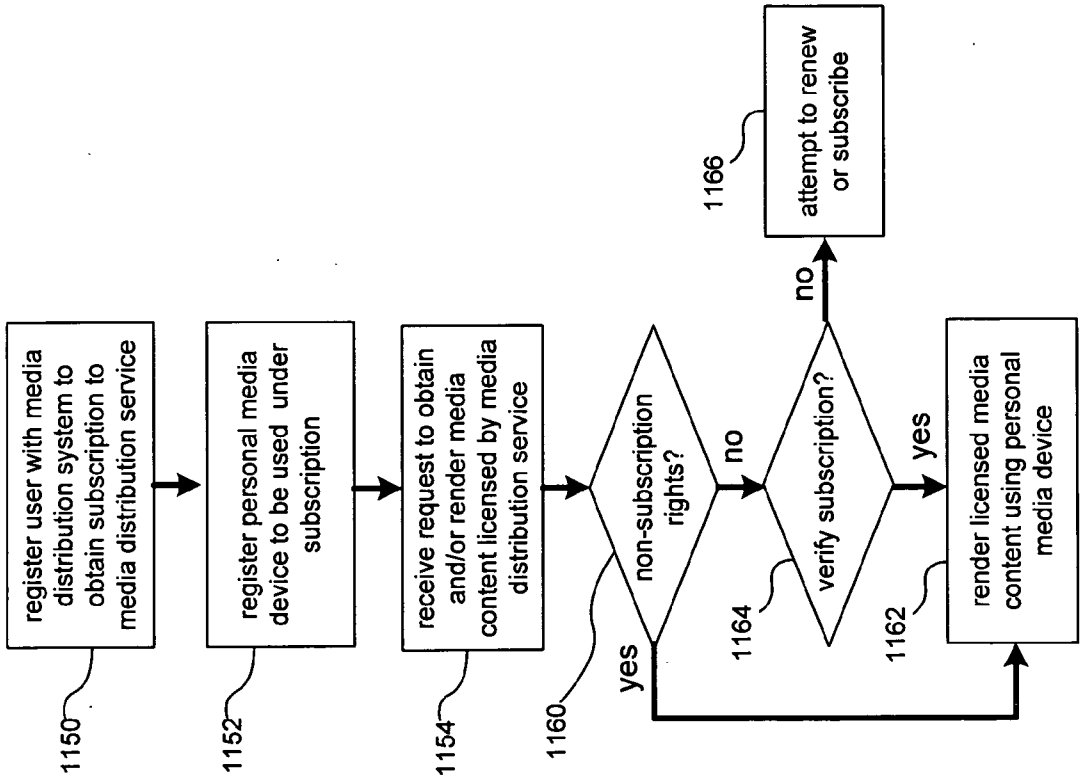

register user with media distribution system to obtain subscription to media distribution service
1150
register personal media device to be used under subscription
1152
receive request to obtain and/or render media content licensed by media distribution service
1154
1160
non-subscription rights?
yes
no
1164
verify subscription?
no
1166
attempt to renew or subscribe
yes
1162
render licensed media content using personal media device

FIG. 17

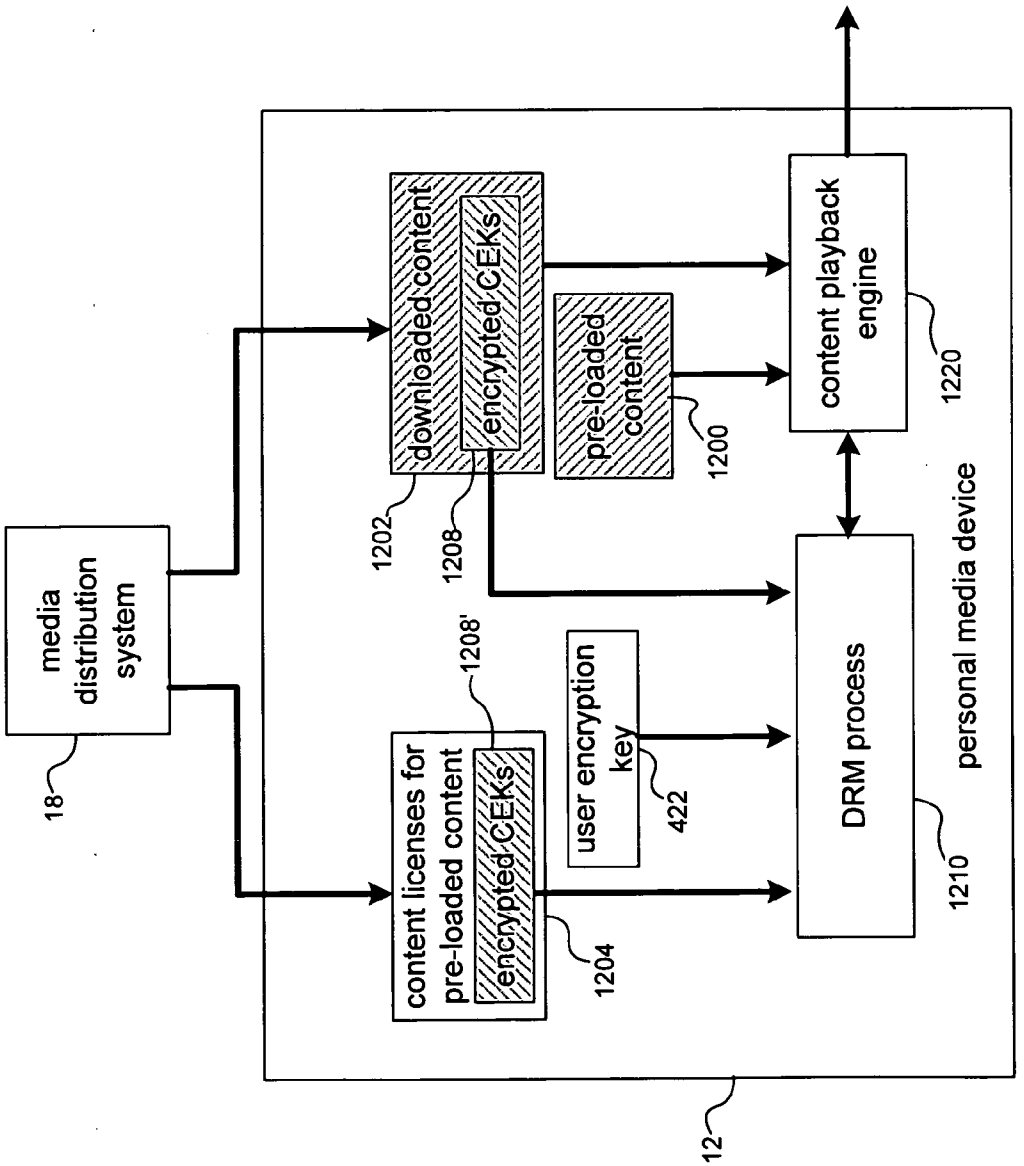

media distribution system
18
downloaded content
encrypted CEKs
1202
1208
pre-loaded content
1200
content playback engine
1220
content licenses for pre-loaded content
encrypted CEKs
1208'
1204
user encryption key
422
DRM process
1210
personal media device
12

FIG. 18

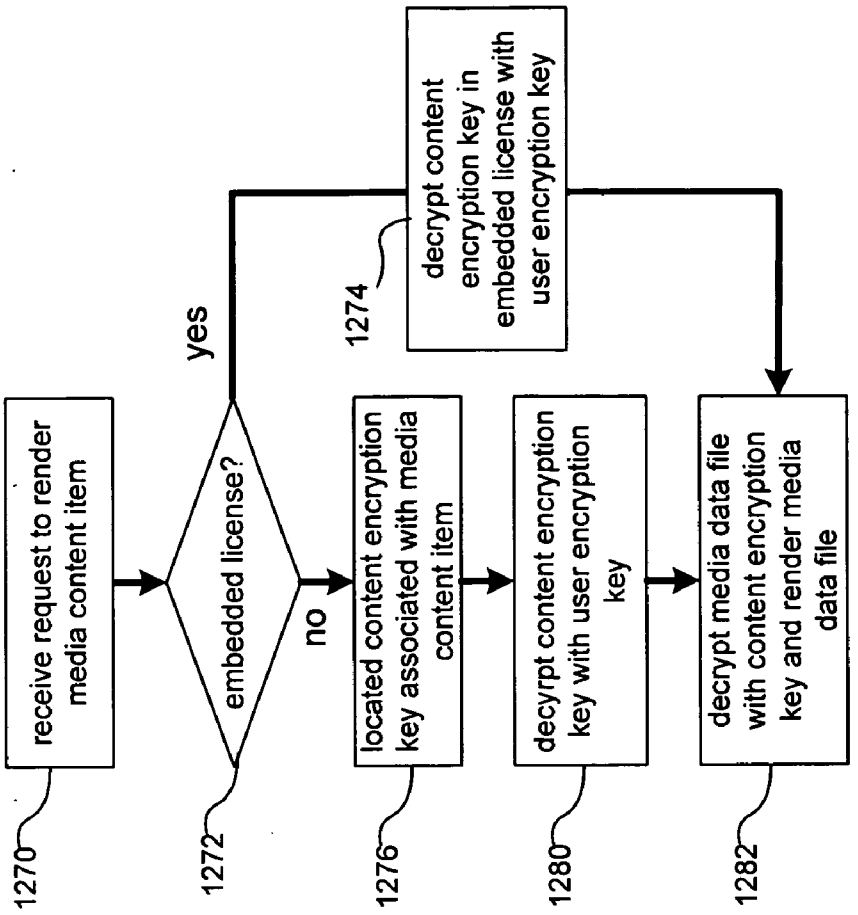

receive request to render media content item
1270
embedded license?
1272
yes
no
decrypt content encryption key in embedded license with user encryption key
1274
located content encryption key associated with media content item
1276
decyrpt content encryption key with user encryption key
1280
decrypt media data file with content encryption key and render media data file
1282

FIG. 20

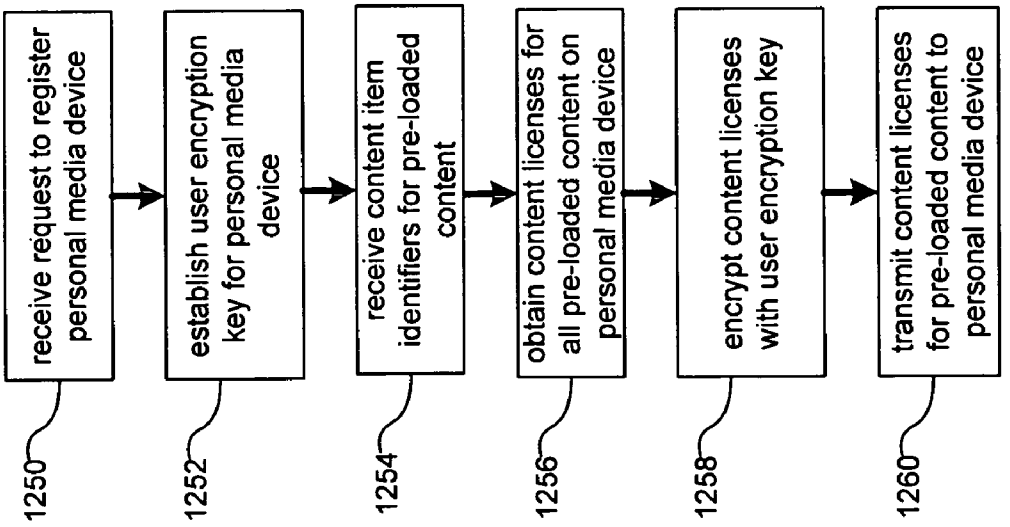

receive request to register personal media device
1250
establish user encryption key for personal media device
1252
receive content item identifiers for pre-loaded content
1254
obtain content licenses for all pre-loaded content on personal media device
1256
encrypt content licenses with user encryption key
1258
transmit content licenses for pre-loaded content to personal media device
1260

FIG. 19

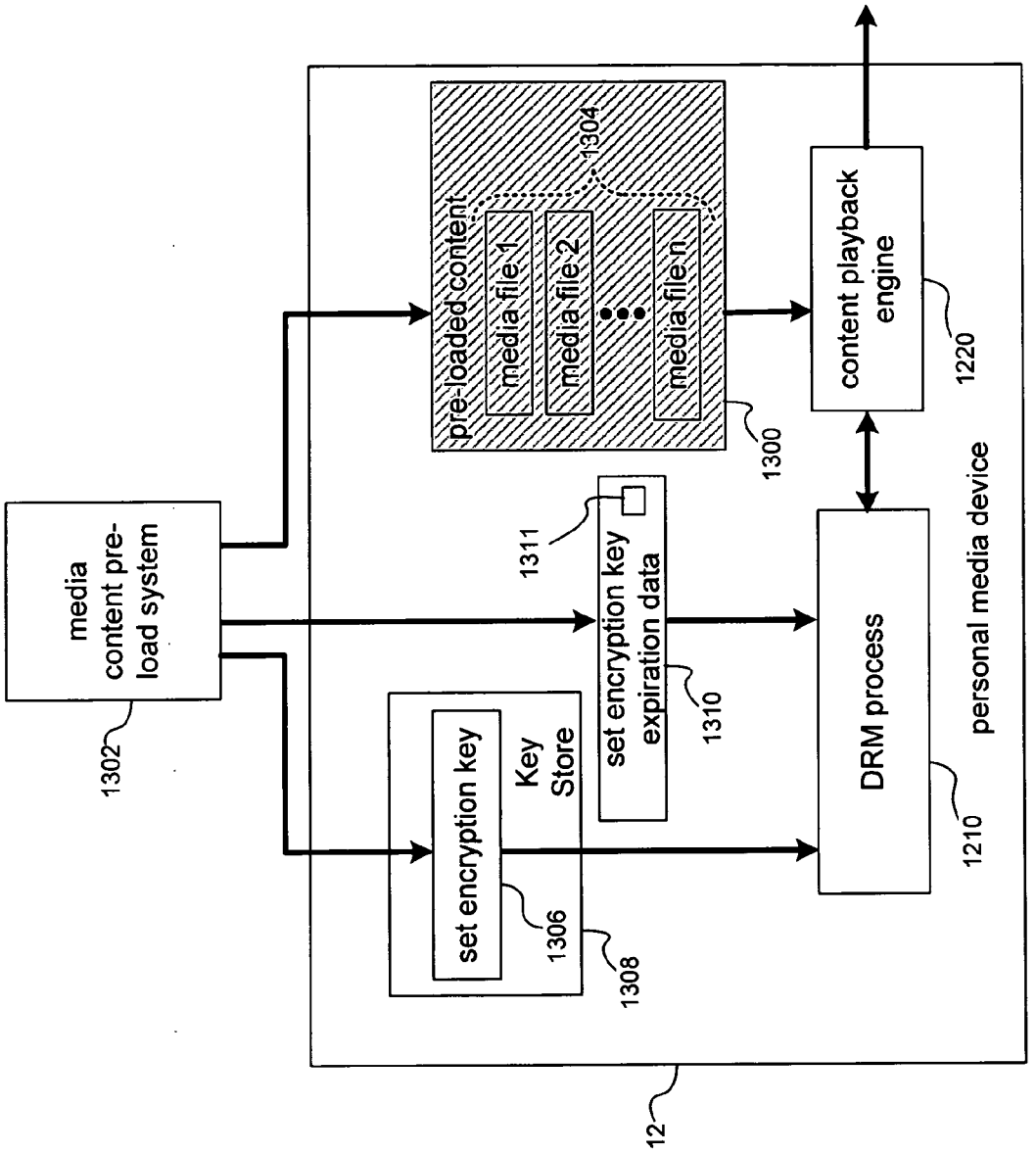

media content pre-load system
1302
pre-loaded content
media file-1
media file-2
media file-n
1304
1300
content playback engine
1220
1311
set encryption key expiration data
1310
DRM process
1210
personal media device
Key Store
set encryption key
1306
1308
12

FIG. 21

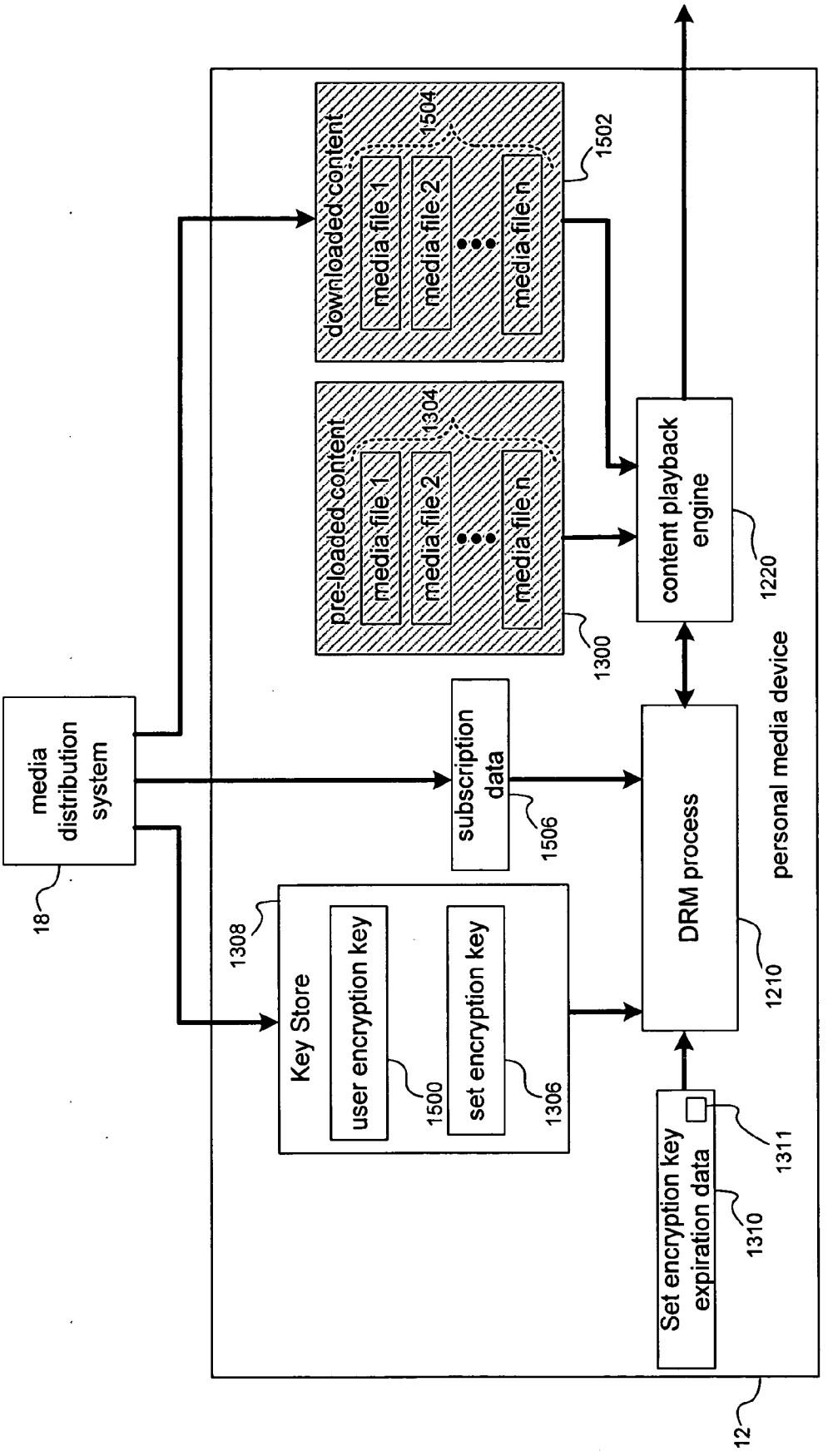

media distribution system
18
downloaded content
media file 1
media file 2
media file n
1504
1502
pre-loaded content
media file 1
media file 2
media file n
1304
1300
content playback engine
1220
subscription data
1506
Key Store
user encryption key
1500
set encryption key
1306
1308
DRM process
1210
personal media device
Set encryption key expiration data
1310
1311
12

FIG. 22

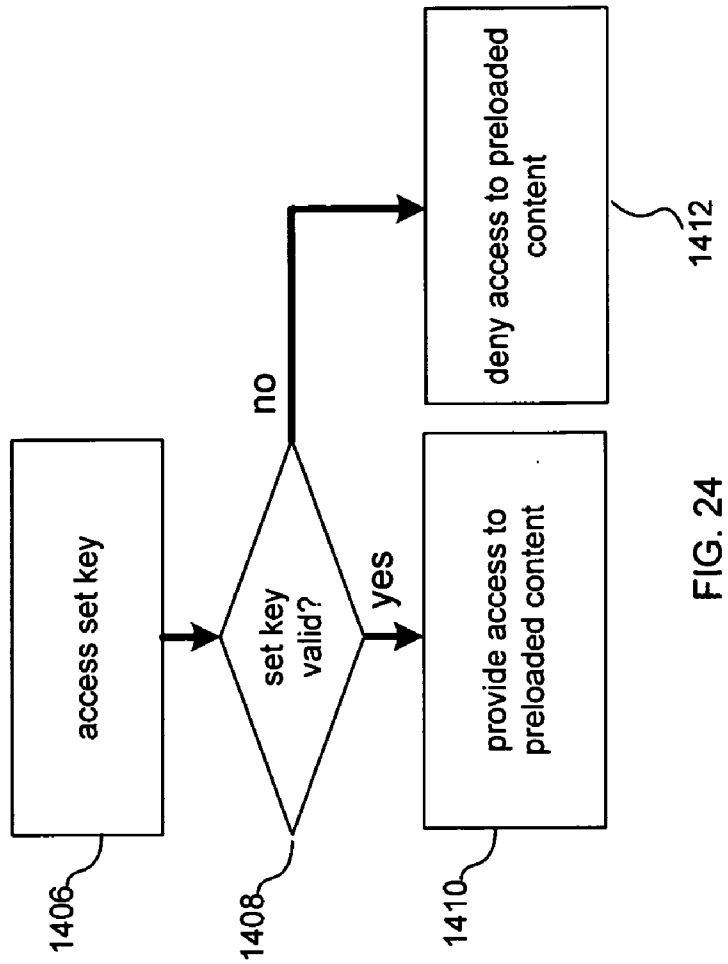

access set key
1406
set key valid?
1408
no
yes
deny access to preloaded content
1412
provide access to preloaded content
1410

FIG. 24

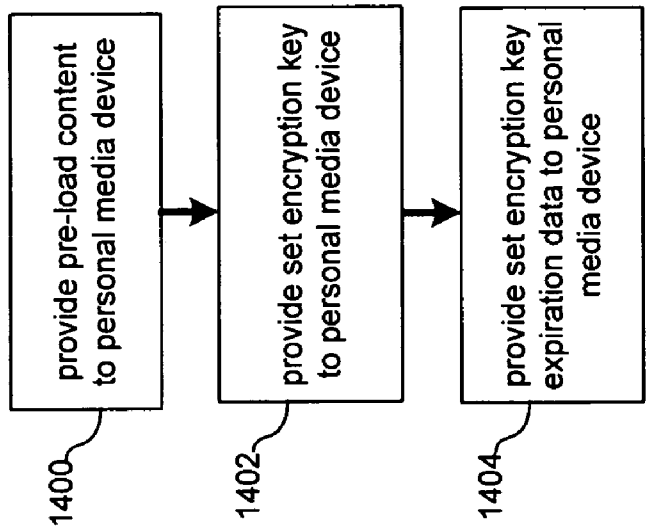

provide pre-load content to personal media device
1400
provide set encryption key to personal media device
1402
provide set encryption key expiration data to personal media device
1404

FIG. 23

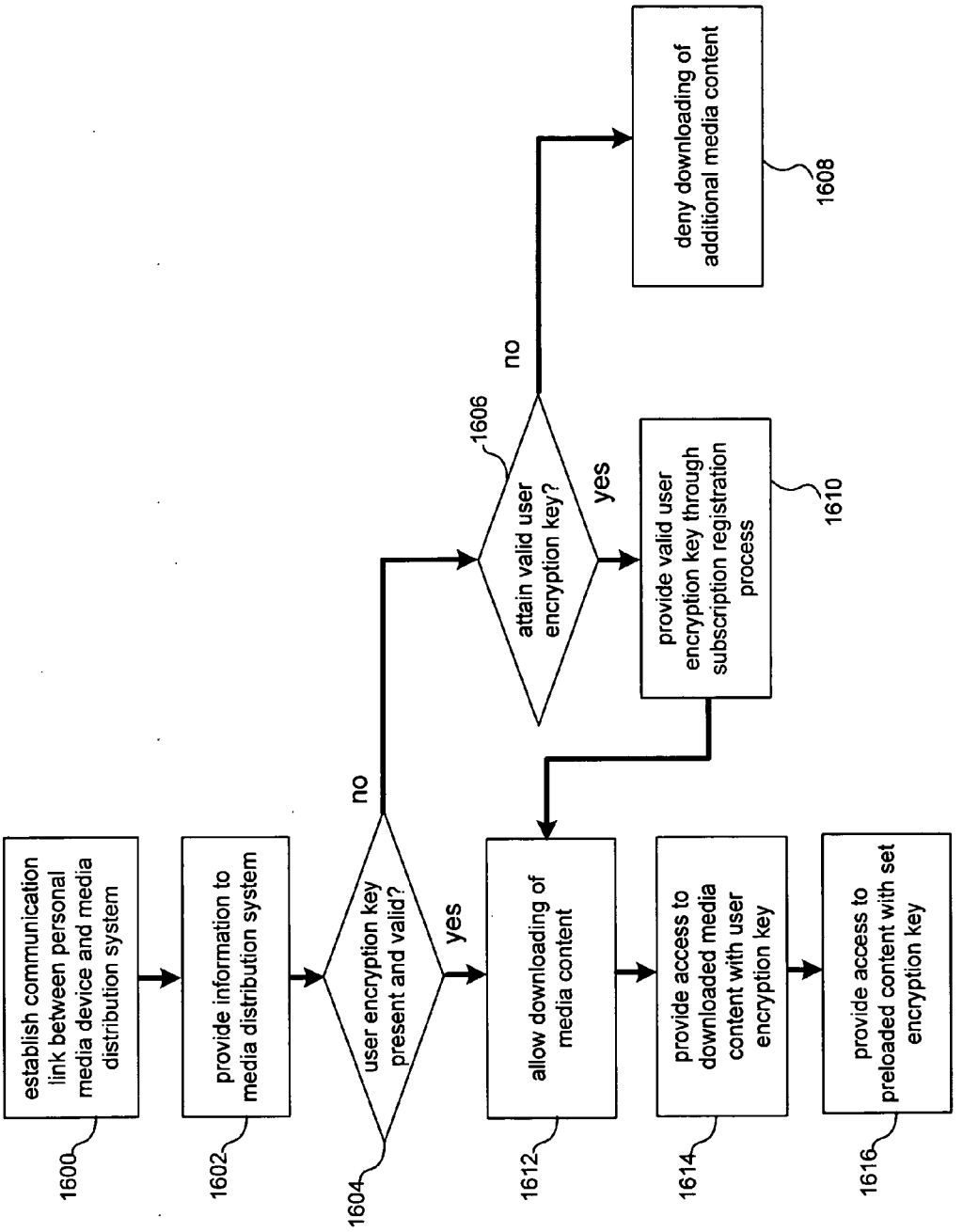

establish communication link between personal media device and media distribution system
1600
provide information to media distribution system
1602
user encryption key present and valid?
1604
yes
no
allow downloading of media content
1612
provide access to downloaded media content with user encryption key
1614
provide access to preloaded content with set encryption key
1616
attain valid user encryption key?
1606
yes
no
provide valid user encryption key through subscription registration process
1610
deny downloading of additional media content
1608

FIG. 25

SYSTEM AND METHOD FOR PRE-LOADING PERSONAL MEDIA DEVICE CONTENT

[0001] This application claims the benefit of priority to U.S. Provisional Application No.: 60/705,969, filed 5 Aug. 2005, entitled "Systems and Methods for Using Personal Media Device".

BACKGROUND

[0002] This invention relates to personal media devices and, more particularly, to pre-loading media content onto the personal media devices.

[0003] Along with at home and office use, people are interested in listening to music while on the move, such as when commuting to and from work. To provide the music, personal media devices (such as the Creative Zen V Plus made by Creative Technology Ltd., of Singapore) may store a relatively large amount of music files and other types of media content (e.g., video content, digital data, etc.). With large storage capacity, a person may store an entire music collection or an entire music genre on one media device. However, a significant amount of time may be needed to download a complete music collection.

SUMMARY

[0004] In one implementation, a method, system and article allows access to pre-loaded media content on a personal media device. Access is allowed depending upon the presence of a pre-loaded content encryption key on the personal media device. The pre-loaded encryption key is associated with the pre-loaded media content.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] FIG. **1** is a diagrammatic view of a DRM process, a media distribution system, a client application, a proxy application, and a personal media device coupled to a distributed computing network;

[0006] FIG. **2** is an isometric view of the personal media device of FIG. **1**;

[0007] FIG. **3** is a diagrammatic view of the personal media device of FIG. **1**;

[0008] FIGS. **4-11** are user interface screens rendered by the client application of FIG. **1**;

[0009] FIG. **12** is a diagrammatic view of the media distribution system, personal media device, and distributed computing network of FIG. **1**;

[0010] FIG. **13** is a diagrammatic view of the media distribution system, personal media device, and distributed computing network of FIG. **1**;

[0011] FIG. **14** is a diagrammatic view of the two personal media devices coupled to each other with a secure communication channel;

[0012] FIG. **15** is a diagrammatic view of an asymmetric key block;

[0013] FIG. **16** is a diagrammatic view of a system for subscription based digital rights management (DRM) on a personal media device;

[0014] FIG. **17** is a flow chart illustrating a method of subscription based digital rights management on a personal media device;

[0015] FIG. **18** is a diagrammatic view of a system for bulk licensing pre-loaded content on a personal media device;

[0016] FIG. **19** is a flow chart illustrating a method of bulk licensing pre-loaded content;

[0017] FIG. **20** is a flow chart illustrating a method of rendering pre-loaded content;

[0018] FIG. **21** is a diagrammatic view of a system for pre-loading content on a personal media device;

[0019] FIG. **22** is a diagrammatic view of a system for loading content onto a personal media device that is storing pre-loaded content;

[0020] FIG. **23** is a flow chart illustrating a method of pre-loading content on a personal media device;

[0021] FIG. **24** is a flow chart illustrating a method of providing access to pre-loaded content on a personal media device; and

[0022] FIG. **25** is a flow chart illustrating a method of downloading and accessing content a personal media that is storing pre-loaded content.

DETAILED DESCRIPTION

[0023] Referring to FIG a DRM (i.e., digital rights management) process **10** that is resident on and executed by personal media device **12** is shown. As will be discussed below in greater detail, DRM process **10** allows a user (e.g., user **14**) of personal media device **12** to manage media content resident on personal media device **12**. Personal media device **12** typically receives media content **16** from media distribution system **18**.

[0024] Examples of the format of the media content **16** received from media distribution system **18** may include: purchased downloads received from media distribution system **18** (i.e., media content licensed to e.g., user **14** for use in perpetuity); subscription downloads received from media distribution system **18** (i.e., media content licensed to e.g., user **14** for use while a valid subscription exists with media distribution system **18**); and media content streamed from media distribution system **18**, for example. Typically, when media content is streamed from, e.g., computer **28** to personal media device **12**, a copy of the media content is not permanently retained on personal media device **12**. In addition to media distribution system **18**, media content may be obtained from other sources, examples of which may include but are not limited to files copied from music compact discs.

[0025] Examples of the types of media content **16** distributed by media distribution system **18** include: audio files (examples of which may include but are not limited to music files, audio news broadcasts, audio sports broadcasts, and audio recordings of books, for example); video files (examples of which may include but are not limited to video footage that does not include sound, for example); audio/video files (examples of which may include but are not limited to a/v news broadcasts, a/v sports broadcasts, feature-length movies and movie clips, music videos, and

episodes of television shows, for example); and multimedia content (examples of which may include but are not limited to interactive presentations and slideshows, for example).

[0026] Media distribution system **18** typically provides media data streams and/or media data files to a plurality of users (e.g., users **14**, **20**, **22**, **24**, **26**). Examples of such a media distribution system **18** include the Rhapsody™ service and Rhapsody-To-Go™ service offered by RealNetworks™ of Seattle, Wash.

[0027] Media distribution system **18** is typically a server application that resides on and is executed by computer **28** (e.g., a server computer) that is connected to network **30** (e.g., the Internet). Computer **28** may be a web server running a network operating system, examples of which may include but are not limited to Microsoft Windows 2000 Server™, Novell Netware™, or Redhat Linux™.

[0028] Typically, computer **28** also executes a web server application, examples of which may include but are not limited to Microsoft IIS™, Novell Webserver™, or Apache Webserver™, that allows for HTTP (i.e., HyperText Transfer Protocol) access to computer **28** via network **30**. Network **30** may be connected to one or more secondary networks (e.g., network **32**), such as: a local area network; a wide area network; or an intranet, for example.

[0029] The instruction sets and subroutines of media distribution system **18**, which are typically stored on a storage device **34** coupled to computer **28**, are executed by one or more processors (not shown) and one or more memory architectures (not shown) incorporated into computer **28**. Storage device **34** may include but are not limited to a hard disk drive, a tape drive, an optical drive, a RAID array, a random access memory (RAM), or a read-only memory (ROM).

[0030] Users **14**, **20**, **22**, **24**, **26** may access media distribution system **18** directly through network **30** or through secondary network **32**. Further, computer **28** (i.e., the computer that executes media distribution system **18**) may be connected to network **30** through secondary network **32**, as illustrated with phantom link line **36**.

[0031] Users **14**, **20**, **22**, **24**, **26** may access media distribution system **18** through various client electronic devices, examples of which may include but are not limited to personal media devices **12**, **38**, **40**, **42**, client computer **44**, personal digital assistants (not shown), cellular telephones (not shown), televisions (not shown), cable boxes (not shown), internet radios (not shown), or dedicated network devices (not shown), for example.

[0032] The various client electronic devices may be directly or indirectly coupled to network **30** (or network **32**). For example, client computer **44** is shown directly coupled to network **30** via a hardwired network connection. Further, client computer **44** may execute a client application **46** (examples of which may include but are not limited to Microsoft Internet Explorer™, Netscape Navigator™, RealRhapsody™ client, RealPlayer™ client, or a specialized interface) that allows e.g., user **22** to access and configure media distribution system **18** via network **30** (or network **32**). Client computer **44** may run an operating system, examples of which may include but are not limited to Microsoft Windows™, or Redhat Linux™.

[0033] The instruction sets and subroutines of client application **46**, which are typically stored on a storage device **48** coupled to client computer **44**, are executed by one or more processors (not shown) and one or more memory architectures (not shown) incorporated into client computer **44**. Storage device **48** may include but are not limited to a hard disk drive, a tape drive, an optical drive, a RAID array, a random access memory (RAM), or a read-only memory (ROM).

[0034] As discussed above, the various client electronic devices may be indirectly coupled to network **30** (or network **32**). For example, personal media device **38** is shown wireless coupled to network **30** via a wireless communication channel **50** established between personal media device **38** and wireless access point (i.e., WAP) **52**, which is shown directly coupled to network **30**. WAP **52** may be, for example, an IEEE 802.11a, 802.11b, 802.11g, Wi-Fi, and/or Bluetooth device that is capable of establishing the secure communication channel **50** between personal media device **38** and WAP **52**. As is known in the art, all of the IEEE 802.11x specifications use Ethernet protocol and carrier sense multiple access with collision avoidance (i.e., CSMA/CA) for path sharing. The various 802.11x specifications may use phase-shift keying (i.e., PSK) modulation or complementary code keying (i.e., CCK) modulation, for example. As is known in the art, Bluetooth is a telecommunications industry specification that allows e.g., mobile phones, computers, and personal digital assistants to be interconnected using a short-range wireless connection.

[0035] In addition to being wirelessly coupled to network **30** (or network **32**), personal media devices may be coupled to network **30** (or network **32**) via a proxy computer (e.g., proxy computer **54** for personal media device **12**, proxy computer **56** for personal media device **40**, and proxy computer **58** for personal media device **42**, for example).

Personal Media Device

[0036] Referring also to FIG. **2**, personal media device **12** may be connected to proxy computer **54** via a docking cradle **60**. Typically, personal media device **12** includes a bus interface (to be discussed below in greater detail) that couples personal media device **12** to docking cradle **60**. Docking cradle **60** may be coupled (with cable **62**) to e.g., a universal serial bus (i.e., USB) port, a serial port, or an IEEE 1394 (i.e., FireWire) port included within proxy computer **54**. For example, the bus interface included within personal media device **12** may be a USB interface, and docking cradle **60** may function as a USB hub (i.e., a plug-and-play interface that allows for "hot" coupling and uncoupling of personal media device **12** and docking cradle **60**).

[0037] Proxy computer **54** may function as an Internet gateway for personal media device **12**. Accordingly, personal media device **12** may use proxy computer **54** to access media distribution system **18** via network **30** (and network **32**) and obtain media content **16**. Specifically, upon receiving a request for media distribution system **18** from personal media device **12**, proxy computer **54** (acting as an Internet client on behalf of personal media device **12**), may request the appropriate web page/service from computer **28** (i.e., the computer that executes media distribution system **18**). When the requested web page/service is returned to proxy computer **54**, proxy computer **54** relates the returned web

page/service to the original request (placed by personal media device **12**) and forwards the web page/service to personal media device **12**. Accordingly, proxy computer **54** may function as a conduit for coupling personal media device **12** to computer **28** and, therefore, media distribution system **18**.

[**0038**] Further, personal media device **12** may execute a device application **64** (examples of which may include but are not limited to RealRhapsody™ client, RealPlayer™ client, or a specialized interface). Personal media device **12** may run an operating system, examples of which may include but are not limited to Microsoft Windows CE™, Redhat Linux™, Palm OS™, or a device-specific (i.e., custom) operating system.

[**0039**] DRM process **10** is typically a component of device application **64** (examples of which may include but are not limited to an embedded feature of device application **64**, a software plug-in for device application **64**, or a stand-alone application called from within and controlled by device application **64**). The instruction sets and subroutines of device application **64** and DRM process **10**, which are typically stored on a storage device **66** coupled to personal media device **12**, are executed by one or more processors (not shown) and one or more memory architectures (not shown) incorporated into personal media device **12**. Storage device **66** may be, for example, a hard disk drive, an optical drive, a random access memory (RAM), a read-only memory (ROM), a CF (i.e., compact flash) card, an SD (i.e., secure digital) card, a SmartMedia card, a Memory Stick, and a MultiMedia card, for example.

[**0040**] An administrator **68** typically accesses and administers media distribution system **18** through a desktop application **70** (examples of which may include but are not limited to Microsoft Internet Explorer™, Netscape Navigator™, or a specialized interface) running on an administrative computer **72** that is also connected to network **30** (or network **32**).

[**0041**] The instruction sets and subroutines of desktop application **70**, which are typically stored on a storage device (not shown) coupled to administrative computer **72**, are executed by one or more processors (not shown) and one or more memory architectures (not shown) incorporated into administrative computer **72**. The storage device (not shown) coupled to administrative computer **72** may include but are not limited to a hard disk drive, a tape drive, an optical drive, a RAID array, a random access memory (RAM), or a read-only memory (ROM).

[**0042**] Referring to FIG. **3**, a diagrammatic view of personal media device **12** is shown. Personal media device **12** typically includes microprocessor **150**, non-volatile memory (e.g., read-only memory **152**), and volatile memory (e.g., random access memory **154**); each of which is interconnected via one or more data/system buses **156**, **158**. Personal media device **12** may also include an audio subsystem **160** for providing e.g., an analog audio signal to an audio jack **162** for removable engaging e.g., a headphone assembly **164**, a remote speaker assembly **166**, or an ear bud assembly **168**, for example. Alternatively, personal media device **12** may be configured to include one or more internal audio speakers (not shown).

[**0043**] Personal media device **12** may also include a user interface **170** and a display subsystem **172**. User interface **170** may receive data signals from various input devices included within personal media device **12**, examples of which may include (but are not limited to): rating switches **74**, **76**; backward skip switch **78**; forward skip switch **80**; play/pause switch **82**; menu switch **84**; radio switch **86**; and slider assembly **88**, for example. Display subsystem **172** may provide display signals to display panel **90** included within personal media device **12**. Display panel **90** may be an active matrix liquid crystal display panel, a passive matrix liquid crystal display panel, or a light emitting diode display panel, for example.

[**0044**] Audio subsystem **160**, user interface **170**, and display subsystem **172** may each be coupled with microprocessor **150** via one or more data / system buses **174**, **176**, **178** (respectively).

[**0045**] During use of personal media device **12**, display panel **90** may be configured to display e.g., the title and artist of various pieces of media content **92**, **94**, **96** stored within personal media device **12**. Slider assembly **88** may be used to scroll upward or downward through the list of media content stored within personal media device **12**. When the desired piece of media content is highlighted (e.g., "Phantom Blues" by "Taj Mahal"), user **14** may select the media content for rendering using play/pause switch **82**. User **14** may skip forward to the next piece of media content (e.g., "Happy To Be Just . . . " by "Robert Johnson") using forward skip switch **80**; or skip backward to the previous piece of media content (e.g., "Big New Orleans . . . " by "Leroy Brownstone") using backward skip switch **78**. Additionally, user **14** may rate the media content as they listen to it by using rating switches **74**, **76**.

[**0046**] As discussed above, personal media device **12** may include a bus interface **180** for interfacing with e.g., proxy computer **54** via docking cradle **60**. Additionally and as discussed above, personal media device **12** may be wireless coupled to network **30** via a wireless communication channel **50** established between personal media device **12** and e.g., WAP **52**. Accordingly, personal media device **12** may include a wireless interface **182** for wirelessly-coupling personal media device **12** to network **30** (or network **32**) and/or other personal media devices. Wireless interface **182** may be coupled to an antenna assembly **184** for RF communication to e.g., WAP **52**, and/or an IR (i.e., infrared) communication assembly **186** for infrared communication with e.g., a second personal media device (such as personal media device **40**). Further and as discussed above, personal media device **12** may include a storage device **66** for storing the instruction sets and subroutines of device application **64** and DRM process **10**. Additionally, storage device **66** may be used to store media data files downloaded from media distribution system **18** and to temporarily store media data streams (or portions thereof) streamed from media distribution system **18**.

[**0047**] Storage device **66**, bus interface **180**, and wireless interface **182** may each be coupled with microprocessor **150** via one or more data/system buses **188**, **190**, **192** (respectively).

[**0048**] As discussed above, media distribution system **18** distributes media content to users **14**, **20**, **22**, **24**, **26**, such that the media content distributed may be in the form of media data streams and/or media data files.

[**0049**] Accordingly, media distribution system **18** may be configured to only allow users to download media data files.

For example, user **14** may be allowed to download, from media distribution system **18**, media data files (i.e., examples of which may include but are not limited to MP3 files or AAC files), such that copies of the media data file are transferred from computer **28** to personal media device **12** (being stored on storage device **66**).

[**0050**] Alternatively, media distribution system **18** may be configured to only allow users to receive and process media data streams of media data files. For example, user **22** may be allowed to receive and process (on client computer **44**) media data streams received from media distribution system **18**. As discussed above, when media content is streamed from e.g., computer **28** to client computer **44**, a copy of the media data file is not permanently retained on client computer **44**.

[**0051**] Further, media distribution system **18** may be configured to allow users to receive and process media data streams and download media data files. Examples of such a media distribution system include the Rhapsody™ and Rhapsody-to-Go™ services offered by RealNetworks™ of Seattle, Wash. Accordingly, user **14** may be allowed to download media data files and receive and process media data streams from media distribution system **18**. Therefore, copies of media data files may be transferred from computer **28** to personal media device **12** (i.e., the received media data files being stored on storage device **66**); and streams of media data files may be received from computer **28** by personal media device **12** (i.e., with portions of the received stream temporarily being stored on storage device **66**). Additionally, user **22** may be allowed to download media data files and receive and process media data streams from media distribution system **18**. Therefore, copies of media data files may be transferred from computer **28** to client computer **44** (i.e., the received media data files being stored on storage device **48**); and streams of media data files may be received from computer **28** by client computer **44** (i.e., with portions of the received streams temporarily being stored on storage device **48**).

[**0052**] Typically, in order for a device to receive and process a media data stream from e.g., computer **28**, the device must have an active connection to computer **28** and, therefore, media distribution system **18**. Accordingly, personal media device **38** (i.e., actively connected to computer **28** via wireless channel **50**), and client computer **44** (i.e., actively connected to computer **28** via a hardwired network connection) may receive and process media data streams from e.g., computer **28**.

[**0053**] As discussed above, proxy computers **54**, **56**, **58** may function as a conduit for coupling personal media devices **12**, **40**, and **42** (respectively) to computer **28** and, therefore, media distribution system **18**. Accordingly, when personal media devices **12**, **40**, and **42** are coupled to proxy computers **54**, **56**, and **58** (respectively) via e.g., docking cradle **60**, personal media devices **12**, **40**, and **42** are actively connected to computer **28** and, therefore, may receive and process media data streams provided by computer **28**.

User Interfaces

[**0054**] As discussed above, media distribution system **18** may be accessed using various types of client electronic devices, which include but are not limited to personal media devices **12**, **38**, **40**, **42**, client computer **44**, personal digital assistants (not shown), cellular telephones (not shown), televisions (not shown), cable boxes (not shown), internet radios (not shown), or dedicated network devices (not shown), for example. Typically, the type of interface used by the user (when configuring media distribution system **18** for a particular client electronic device) will vary depending on the type of client electronic device to which the media content is being streamed/downloaded.

[**0055**] For example, as the embodiment shown (in FIG. **2**) of personal media device **12** does not include a keyboard and the display panel **90** of personal media device **12** is compact, media distribution system **18** may be configured for personal media device **12** via proxy application **98** executed on proxy computer **54**.

[**0056**] The instruction sets and subroutines of proxy application **98**, which are typically stored on a storage device (not shown) coupled to proxy computer **54**, are executed by one or more processors (not shown) and one or more memory architectures (not shown) incorporated into proxy computer **54**. The storage device (not shown) coupled to proxy computer **54** may include but are not limited to a hard disk drive, a tape drive, an optical drive, a RAID array, a random access memory (RAM), or a read-only memory (ROM).

[**0057**] Additionally and for similar reasons, personal digital assistants (not shown), cellular telephones (not shown), televisions (not shown), cable boxes (not shown), internet radios (not shown), and dedicated network devices (not shown) may use proxy application **98** executed on proxy computer **54** to configure media distribution system **18**.

[**0058**] Further, the client electronic device need not be directly connected to proxy computer **54** for media distribution system **18** to be configured via proxy application **98**. For example, assume that the client electronic device used to access media distribution system **18** is a cellular telephone. While cellular telephones are typically not physically connectable to e.g., proxy computer **54**, proxy computer **54** may still be used to remotely configure media distribution system **18** for use with the cellular telephone. Accordingly, the configuration information (concerning the cellular telephone) that is entered via e.g., proxy computer **54** may be retained within media distribution system **18** (on computer **28**) until the next time that the user accesses media distribution system **18** with the cellular telephone. At that time, the configuration information saved on media distribution system **18** may be downloaded to the cellular telephone.

[**0059**] For systems that include keyboards and larger displays (e.g., client computer **44**), client application **46** may be used to configure media distribution system **18** for use with client computer **44**.

[**0060**] Referring now to FIG. **4**, when using client application **46** to access media distribution system **18**, user **22** may be presented with an information display screen **200** rendered by client application **46**. Client application **46** typically includes a user interface **202** (e.g., a web browser) for interfacing with media distribution system **18** and viewing information display screen **200**.

[**0061**] When e.g., user **22** streams/downloads media content from e.g., computer **28**, media distribution system **18** may monitor the media content streamed/downloaded to the user's client electronic device (e.g., client computer **44**, for example), resulting in the generation of a media history file

100 for that user. While media history file **100** is typically maintained locally (e.g., maintained on client computer **44**), media history file **100** may alternatively/additionally be maintained remotely (e.g., maintained on computer **28**) as a remote media history file **100'**.

[**0062**] The user (e.g., user **22**) may save this media history file (or portions thereof) as a playlist. A playlist is typically a group of tracks (examples of which may include, but are not limited to, songs, videos, news broadcasts, sports broadcasts, etc) that media distribution system **18** will render in sequence. This, in turn, allows the user to compile custom music compilations (in the form of multiple playlists).

[**0063**] A history window **204** may be rendered by client application **46** that itemizes the information contained within media history file **100**. In this example, history window **204** itemizes ten (**10**) media data streams (e.g., "Jailhouse Rock"; "Surf City"; "Runaround Sue"; "The Wanderer"; "The Great Pretender"; "Blueberry Hill"; "I'm Walkin"; "Blue Christmas"; "Yakety Yak"; and "Peggy Sue"), thus indicating that user **22** had previously listened to those ten (10) media data streams.

[**0064**] In addition to media data streams (i.e., media data streams received from a remote device e.g., computer **28**), client application **46** allows user **12** to render local media data files. As discussed above, a local media data file may be a purchased download received from media distribution system **18** (i.e., media content licensed to e.g., user **14** for use in perpetuity); a subscription download received from media distribution system **18** (i.e., media content licensed to e.g., user **14** for use while a valid subscription exists with media distribution system **18**); and/or a media data file extracted (i.e., ripped) from e.g., a music compact disc, for example. These local media data files are typically stored locally on e.g., storage device **48** coupled to client computer **44**.

[**0065**] If user **22** wishes to render a local media data file (i.e., a file stored on client computer **44**), user **22** may e.g., select the file(s) to be rendered using client application **46**. Accordingly, user **22** may select the dropdown "File" menu **206** using screen pointer **208**, which is controllable by a pointing device (e.g., a computer mouse, not shown). Selecting the "Open" command may result in client application **46** rendering file management window **210**, which allows user **22** to select local media data files for playback.

[**0066**] In this example, file management window **210** defines three (**3**) local media data files, namely: "Chantilly Lace"**212**; "Great Balls of Fire"**214**; and "Tutti Frutti"**216**, all of which are stored within the folder "My Music". User **22** may select any (or all) of these files for playback on client application **46**.

[**0067**] A search window **218** allows a user (e.g., user **22**) to search for media content. For example, user **22** may enter search terms (e.g., "Elvis Presley"), select the appropriate term type (e.g., artist), and execute a query. In the event that multiple artists satisfy the query, a result set is generated from which user **22** may select e.g., the appropriate artist. Once the appropriate artist is selected, user **22** may review the various albums released by the selected artist (or that include tracks by the selected artist). User **22** may then render one or more of the various tracks included within any of the albums. Once a track is rendered, identifying information concerning the track rendered is added to local media history file **100** and/or remote media history file **100'** and is included in history window **204**. In addition to being able to search for media content by artist, user **14** may also be able to search for media content by e.g., keyword, track, album and/or composer.

[**0068**] Referring now to FIG. **5** and assuming that user **22** selects all three local media data files for playback, media history file **100** is amended to include three additional entries, namely one for "Chantilly Lace"; one for "Great Balls of Fire"; and one for "Tutti Frutti". Accordingly, as history window **204** itemizes the information contained within media history file **100**, history window **204** will include three additional entries (i.e., entries **220**, **222**, **224**), which correspond to local media data file "Chantilly Lace"**212**; local media data file "Great Balls of Fire"**214**; and local media data file "Tutti Frutti"**216**.

[**0069**] Assuming that user **22** wishes to save this collection of music for future playback, user **22** may save the current media history file **100** (or a portion thereof) as a playlist **102** (FIG. **1**). While playlist **102** is typically maintained locally (e.g., maintained on client computer **44**), playlist **102** may alternatively/additionally be maintained remotely (e.g., maintained on computer **28**) as a remote playlist **102'**.

[**0070**] Referring now to FIG. **6**, user **22** may select the "save" button **240** (using screen pointer **208**). Once the "save" button **240** is selected, a playlist naming window **242** is rendered (by client application **46**) that allows user **22** to specify a unique name for playlist **102** within the name field **244** of playlist naming window **242**.

[**0071**] Assuming that user **22** selects "50's Hits" as a playlist name, playlist **102** is saved (i.e., as "50's Hits") and defines the location of all of the pieces of media content itemized within history window **204**.

[**0072**] Referring now to FIG. **7**, once playlist **102** is stored, a link **260** to playlist **102** (e.g., "50's Hits") appears in directory window **262**. User **22** may then select link **260** using screen pointer **208**. Once selected, the tracks included within playlist **102** (e.g., "50's Hits") are itemized within a playlist window **264** (e.g., a web page) viewable via user interface **202**. As discussed above, ten of these entries (namely "Jailhouse Rock"; "Surf City"; "Runaround Sue"; "The Wanderer"; "The Great Pretender"; "Blueberry Hill"; "I'm Walkin"; "Blue Christmas"; "Yakety Yak"; and "Peggy Sue") define the location of media data streams and three of these entries (namely "Tutti Frutti"; "Chantilly Lace"; and "Great Balls of Fire") define the location of media data files.

[**0073**] Typically, playlist window **264** includes hyperlinks that locate (i.e., provide addresses for) the streams/files associated with the individual entries itemized within playlist **102**. This location information is stored within playlist **102**. For example, the following table correlates the track name of an entry in playlist **102** with an address for the stream/file associated with that track name:

Track Name	Address
Jailhouse Rock	www.musicshop.com\songs\jailhouse_rock.ram
Surf City	www.musicshop.com\songs\surf_city.ram
Runaround Sue	www.musicshop.com\songs\runaround_sue.ram
The Wanderer	www.musicshop.com\songs\the_wanderer.ram

-continued

Track Name	Address
The Great Pretender	www.musicshop.com\songs\the_great_pretender.ram
Blueberry Hill	www.musicshop.com\songs\blueberry_hill.ram
I'm Walkin'	www.musicshop.com\songs\im_walkin.ram
Blue Christmas	www.musicshop.com\songs\blue_christmas.ram
Yakety Yak	www.musicshop.com\songs\yakety_yak.ram
Peggy Sue	www.musicshop.com\songs\peggy_sue.ram
Tutti Frutti	c:\my music\tutti_frutti.mp3
Chantilly Lace	c:\my music\chantilly_lace.mp3
Great Balls of Fire	c:\my music\great_balls_of_fire.mp3

[0074] As the first ten entries (namely "Jailhouse Rock"; "Surf City"; "Runaround Sue"; "The Wanderer"; "The Great Pretender"; "Blueberry Hill"; "I'm Walkin"; "Blue Christmas"; "Yakety Yak"; and "Peggy Sue") identify media data streams, the address provided for each entry points to a media stream available from e.g., media distribution system **18**. Further, as the last three entries (namely "Tutti Frutti"; "Chantilly Lace"; and "Great Balls of Fire") identify media data files, the address provided for each entry points to a media data file available from e.g., client computer **44**.

[0075] Playlist window **264** is typically tabular and may include a column **266** identifying a media type (i.e., media data stream or media data file, for example) for each entry within playlist window **264**. Typically, column **266** includes icons that identify the media type (e.g., icon **268** identifies a media data file and icon **270** identifies a media data stream). User **22** may select the "play" button **272** to render playlist **102**.

[0076] As discussed above, media distribution system **18** typically provides media data streams and/or media data files to users (e.g., user **22**). Typically, metadata is associated with each media data stream provided by media distribution system **18**. This metadata may include (but is not limited to) an artist identifier, an album identifier, a track identifier, an album cover image, and a music genre identifier, for example.

[0077] Accordingly, whenever e.g., user **12** renders a remote media data stream, media distribution system **18** may compile and save this metadata (on a per-user basis) to track e.g., listening trends and musical preferences of individual users, for example.

[0078] As discussed above, a local media data file may be a purchased download received from media distribution system **18** (i.e., media content licensed to e.g., user **14** for use in perpetuity); a subscription download received from media distribution system **18** (i.e., media content licensed to e.g., user **14** for use while a valid subscription exists with media distribution system **18**); and/or a media data file extracted (i.e., ripped) from e.g., a music compact disc, for example.

[0079] If the purchased download and/or the subscription download were provided by media distribution system **18**, these local media data files would typically also include the metadata described above. Accordingly, when these purchased/subscription downloads are rendered by e.g., user **22**, the metadata concerning these purchased/subscription downloads may be transmitted from computer **44** to computer **28**, such that the metadata is compiled and saved (on a per user basis) to track e.g., listening trends and musical preferences, for example.

[0080] However, for media data files that were e.g., extracted from music compact discs, these data files may not include the above-described metadata. As discussed above, media data files (i.e., files stored on client computer **44**) may be rendered using client application **46** and added to playlists (e.g., playlist **102**). Accordingly, whenever user **22** attempts to add a media data file (that does not include metadata) to a playlist (e.g., playlist **102**), user **22** may be prompted to provide metadata concerning that media data file.

[0081] Referring also to FIG. **8** and continuing with the above-stated example, if user **22** attempts to save a playlist (e.g., playlist **102**) that includes three local media data files (namely "Tutti Frutti"; "Chantilly Lace"; and "Great Balls of Fire"), assuming that these three local media data files do not include metadata, client application **46** may render a metadata entry form **280** that allows user **22** to enter metadata concerning each of the three media data files.

[0082] In this example, metadata entry form **280** includes five user-editable fields, namely an artist field **282**, an album field **284**, a track field **286**, an album cover image field **288**, and a music genre field **290**. Album cover image field **288** may allow user **22** to define a drive, a path, and a filename for an album cover image. Music genre field **290** may be a drop-down menu (operable via screen pointer **208**) that allows user **22** to select a music genre from a number of predefined music genres (not shown).

[0083] Typically, if the title of the media data file is descriptive of the track name, the track field **286** may be automatically-populated with what client application **46** suspects is the track title. As the first local media data file is named "tutti frutti", track field **286** would typically be populated with the suspected name "tutti frutti". User **22** may populate the remaining fields and select the save button **292** (using screen pointer **208**) or alternatively select the cancel button **294**.

[0084] In order to further automate the metadata generation process, client application **44** may interface with a remote metadata database (not shown) served by e.g., media distribution system **18** or a third party (not shown). This metadata database may define metadata for various tracks and albums. An example of such a database is the CDDB™ database maintained by Gracenote™ of Emeryville, Calif. (www.gracenote.com). For example, if user **22** ripped each track from an entire compact disc, the metadata database may be accessed by client application **44** and a query may be structured that defines e.g., the total number of tracks included on the compact disc, the length of each track included on the compact disc, and the total length of the compact disc. Assuming that a definitive result is produced by this query, the metadata for each track ripped from the compact disc would be produced. In the event that an indefinite result set (i.e., one that identifies multiple possible compact discs) is generated, user **22** may be prompted to select the appropriate compact disc from a list of possible matches (not shown).

[0085] As discussed above, the type of interface used by the user (when configuring media distribution system **18** for

a client electronic device) may vary depending on the type and the capabilities of the client electronic device to which the media content is being streamed/downloaded. Accordingly and as discussed above, media distribution system **18** may be configured for personal media device **12** via proxy application **98** executed on proxy computer **54**.

[**0086**] Proxy application **98** may be automatically executed upon personal media device **12** being placed into docking cradle **60** by e.g., user **14**. Alternatively, proxy application **98** may be fully or partially loaded upon boot up of proxy computer **54**. Proxy application **98** may then operate in the background until personal media device **12** is placed into docking cradle **60**, at which time proxy application **98** may be fully loaded and/or moved to the foreground for execution. Further, proxy application **98** may be manually executed by user **14**. As will be discussed below in greater detail, proxy application **98** (once executed) may be used to e.g., configure personal media device **12** and transfer media data files to and remove media data files from personal media device **12**, for example.

[**0087**] Referring also to FIG. **9**, when using proxy application **98** to access media distribution system **18**, user **14** may be presented with an information display screen **300** rendered by proxy application **98**. Proxy application **98** typically includes a user interface **302** (e.g., a web browser) for interfacing with media distribution system **18** and viewing information display screen **300**.

[**0088**] A search window **304** allows a user (e.g., user **14**) to search for media content. For example, user **14** may enter search terms (e.g., "Elvis Presley") into search field **306**, select the appropriate term type (e.g., artist), and execute a query. In the event that multiple artists satisfy the query, a result set is generated from which user **14** may select e.g., the appropriate artist. Once the appropriate artist is selected, user **14** may review the various albums released by the selected artist (or that include tracks by the selected artist). User **14** may then download (for use on personal media device **12**) one or more of the various tracks included within any of the albums. In addition to being able to search for media content by artist, user **14** may also be able to search for media content by e.g., keyword, track, album and/or composer.

[**0089**] Additionally, in a fashion similar to that of client application **46**, proxy application **98** may be configured to allow user **12** to render (via proxy computer **54**) one or more of the various tracks included within any of the albums of the selected artist.

[**0090**] A content window **308** may be rendered by proxy application **98** that allows user **14** to review the contents of personal media device **12**. As discussed above, personal media device **12** is coupled to proxy computer **54** via e.g., a USB port, serial port, or FireWire port. Upon or during execution of proxy application **98**, proxy application **98** may poll personal media device **12** to retrieve information concerning the media content currently on device **12**. This polling may occur in a fashion similar to the manner in which the content of a USB hard drive is determined. In this particular example, content window **308** includes ten (10) entries, namely: "Jailhouse Rock"; "Surf City"; "Runaround Sue"; "The Wanderer"; "The Great Pretender"; "Blueberry Hill"; "I'm Walkin"; "Blue Christmas"; "Yakety Yak"; and "Peggy Sue", thus indicating that ten (10) media data files had been previously downloaded to personal media device **12**, which are typically stored on storage device **66** of personal media device **12**.

[**0091**] Content window **308** may be tabular and itemize various pieces of information concerning the downloaded files, including the track **310**, the artist **312**, the track length **314** and the track size **316**. Additionally, proxy application **98** may poll personal media device **14** to retrieve device identification information, which is rendered within a device type field **320** and a device serial number field **322** included within content window **308**. Further, content window **308** may include a summary information field **324** concerning the current capacity of device **12**, including one or more of e.g., "Unused Space" in gigabytes; "Used Space" in gigabytes; "Unused Space" in percentage of total capacity; and "Used Space" in percentage of total capacity, for example.

[**0092**] Referring also to FIG. **10** and continuing with the above-stated example, assume that user **14** enters the search term "Elvis Presley" into search field **306** of search window **304**, selects the term type "artist" via dropdown menu **340**, and executes the query by selecting the "Go" button **342** with screen pointer **208**.

[**0093**] Assuming that no other artist satisfies the query, information screen **300** may be presented to user **14** with information concerning Elvis Presley, which may include: an artist information screen **344**, a top track list **346**, an album list **348**, and a similar artist list **350**, for example.

[**0094**] User **14** may download media data files from media distribution system **18** for use on personal media device **12** by selecting the download button **352** corresponding to the track to be downloaded. Additionally, user **14** may download groups of tracks (e.g., each track included within top track list **346**, or all tracks included within an single album) by selecting the download all button **354** corresponding to the tracks to be downloaded.

[**0095**] Once user **14** selects a track for downloading, proxy application **98** may render a download window **356** that e.g., includes a track title field **358** that identifies the title of the track being downloaded and an artist field **360** that identifies the artist of the track being downloaded.

[**0096**] As discussed above, files may be downloaded from media distribution system **18** as purchased downloads (i.e., media content licensed to e.g., user **14** for use in perpetuity), or subscription downloads (i.e., media content licensed to e.g., user **14** for use while a valid subscription exists with media distribution system **18**). Provided user **14** has a current subscription with media distribution system **18**, there is typically no additional fee charged for each subscription download, as the downloaded media content is only renderable while the user has a valid subscription. However, a user typically must pay a fee (e.g., 79¢, 89¢, or 99¢, for example) for each purchased download, as the media content is renderable regardless of the status of the user's subscription.

[**0097**] Accordingly, download window **356** may include a purchase button **362** and a download button **364**, both of which are selectable via screen pointer **208**. In this example, if user **14** selects purchase button **362** with screen pointer **208**, a media data file for "Hound Dog" by "Elvis Presley" will be transferred from computer **28** to personal media device **12**. Typically, user **14** will be charged e.g., a one-time

download fee for downloading this media data file. However, as this is a purchased download, the media data file received is renderable regardless of the status of the user's subscription with media distribution system **18**.

[**0098**] Alternatively, if user **14** selects download button **364** with screen pointer **208**, a media data file for "Hound Dog" by "Elvis Presley" will be transferred from computer **28** to personal media device **12**. Typically, user **14** will not be charged a fee for downloading this media data file. However, as this is a subscription download, the media data file received is only renderable while user **14** has a valid subscription with media distribution system **18**.

[**0099**] Download window **356** typically also includes a cancel button **366** for allowing user **14** to cancel the download and close download window **356**.

[**0100**] If user **14** selects either purchase button **362** or download button **364**, the download of the selected media data file will be initiated. Download window **356** may include a download status indicator **368** for indicating the progress of the download of e.g., "Hound Dog" by "Elvis Presley".

[**0101**] Referring also to FIG. **11**, once the download of the media data file for "Hound Dog" by "Elvis Presley" is completed, content window **308** will be updated to include an entry **380** for "Hound Dog" by "Elvis Presley", indicating that "Hound Dog" by "Elvis Presley" was successfully downloaded from media distribution system **18** to personal media device **12**.

[**0102**] In a fashion similar to that described above concerning client application **46**, user **14** may use proxy application **98** to define playlists concerning various media data files stored on personal media device **12**. For example, assume that user **14** wished to save the first thirteen tracks (namely "Jailhouse Rock"; "Surf City"; "Runaround Sue"; "The Wanderer"; "The Great Pretender"; "Blueberry Hill"; "I'm Walkin"; "Blue Christmas"; "Yakety Yak"; "Peggy Sue"; "Tutti Frutti"; "Chantilly Lace"; and "Great Balls of Fire") as a playlist, user **14** would highlight the desired selection of tracks (using screen pointer **208**) and select the save button **382** using screen pointer **208**. A playlist naming window **384** may be rendered (by proxy application **98**) that allows user **14** to specify a unique name for the playlist within the name field **386** of playlist naming window **384**.

[**0103**] Assuming that user **14** selects "50's Hits" as a playlist name, playlist **104** (FIG. **1**) named "50's Hits" is defined that locates (within personal media device **12**) all of the pieces of media content itemized within playlist **104**. Once playlist **104** is stored, a link **388** to playlist **104** (e.g., "50's Hits") appears in directory window **390**. User **14** may then select link **388** using screen pointer **208**.

[**0104**] Once selected, the tracks included within playlist **104** (e.g., "50's Hits") are typically itemized within a playlist window **392** (e.g., a web page) viewable via user interface **302**.

[**0105**] As with the playlists described above as being generated using client application **44**, playlists generated using proxy application **98** are typically maintained locally (e.g., maintained on personal media device **12**). However and as discussed above, playlists may alternatively/additionally be maintained remotely (e.g., maintained on computer **28**) as remote playlist **104'**.

Device Initialization

[**0106**] Media distribution system **18** is typically a subscription-based service, in that e.g., user **14** subscribes to media distribution system **18** and pays e.g., a monthly subscription fee to be granted access to media distribution system **18**. Once user **14** subscribes to media distribution system **18**, user **14** may obtain media content (for use with personal media device **12**) in the form of: purchased downloads received from media distribution system **18** (i.e., media content licensed to e.g., user **14** for use in perpetuity); subscription downloads received from media distribution system **18** (i.e., media content licensed to e.g., user **14** for use while a valid subscription exists with media distribution system **18**); and media content streamed from media distribution system **18**, for example. Typically, when accessing media distribution system **18**, user **14** must provide user "credentials" that identify the user (e.g., user **14**) and/or the device (e.g., device **12**) to media distribution system **18**. Upon receiving these credentials, media distribution system **18** may attempt to verify the credentials and, if verified, grant user **14** and/or device **12** access to media subscription system **18**. The credentials received and verified by media distribution system **18** may include, but are not limited to, a user name, a user password, a user key, a device name, a device password, a device key, and/or one or more digital certificates.

[**0107**] Typically, upon personal media device **12** being placed into docking cradle **60**, personal media device **12** establishes a connection with media distribution system **18** via proxy computer **54**. As discussed above, proxy computer **54** may function as an Internet gateway for personal media device **12** and, therefore, allow personal media device **12** to access computer **28** and media distribution system **18**.

[**0108**] Once a connection is establish with media distribution system **18**, DRM process **10** may be initiated. DRM process **10** is typically executed at the time personal media device **12** is initially configured (i.e., the first time personal media device **12** establishes a connection with media distribution system **18**). As will be discussed below in greater detail, DRM process **10** may be systematically and repeatedly executed to verify that device **12** (and/or user **14**) are active subscribers of media distribution system **18**.

[**0109**] Referring also to FIG. **12**, at the time of manufacture, personal media device **12** may include a private encryption key (e.g., device private key **400**) and a public encryption key (e.g., device public key **402**) stored in non-volatile memory (e.g., ROM **152** and/or storage device **66**). Keys **400**, **402** may be 1024-bit asymmetric encryption keys and may be referred to as DRM (i.e., digital rights management) keys.

[**0110**] As is known in the art, a private key/public key encryption methodology allows users of an unsecure network (e.g., the Internet) to securely exchange data through the use of a pair of encryption keys, namely the private encryption key (e.g., device private key **400**) and the public encryption key (e.g., device public key **402**). The private key/public key encryption methodology is typically referred to as an asymmetric encryption methodology, in that the key used to encrypt a message is different than the key used to decrypt the message.

[**0111**] In private key/public key encryption, the private encryption key (e.g., device private key **400**) and the public

encryption key (e.g., device public key **402**) are typically created simultaneously using the same algorithm (e.g., the RSA algorithm created by Ron Rivest, Adi Shamir, and Leonard Adlemana, for example). Device private key **400** is typically given only to the requesting party and device public key **402** is typically made publicly available (e.g., as part of digital certificate **404**). Typically, device private key **400** is not shared and is maintained securely within e.g., personal media device **12**.

[**0112**] Accordingly, when a secure message is to be sent from a sender to a recipient, the public key (e.g., device public key **402**) of the recipient (which is readily accessible to the sender) is used to encrypt the message. Once encrypted, the message is sent to the recipient and can only be decrypted using the recipient's private key (e.g., device private key **400**). As private key **400** is maintained securely by the recipient, only the recipient can decrypt the encrypted message.

[**0113**] In addition to encrypting and decrypting messages, a sender may authenticate their identity by using their private key (e.g., device private key **400**) to encrypt a digital certificate, which is then sent to a recipient (i.e., the person to which they are authenticating their identity). Accordingly, when the digital certificate is received by the recipient, the recipient can decrypt the encrypted digital certificate using the sender's public key (e.g., device public key **402**), thus verifying that the digital certificate was encrypted using the sender's private key (e.g., device private key **400**) and, therefore, verifying the identity of the sender.

[**0114**] DRM process **10** may generate a challenge **406**, which is typically a random number generated by a random number generation process (not shown) included within personal media device **12**. Once generated, challenge **406** is paired with device digital certificate **404** (which typically includes device public key **402**), to form a license request **408**. Device digital certificate **404**, which may be referred to as a DRM digital certificate, may include additional information such as a device serial number (e.g., 137660523-1 from device serial number field **322**, FIG. **9**), for example.

[**0115**] As discussed above, proxy application **98** allows the owner of device **12** (e.g., user **14**) to configure device **12** for use with media distribution system **18** and configure media distribution system **18** for use with device **12**. Typically, when proxy application **98** is configured on proxy computer **54**, user **14** may be required to provide user credentials that identify the user (e.g., user **14**) and define a valid subscription that would allow user **14**, device **12**, and proxy application **98** to access media distribution system **18**. Alternatively or additionally, personal media device **12** may be configured to allow the user (e.g., user **14**) to directly enter the user credentials (via device **12**) when device **12** is initially configured.

[**0116**] DRM process **10** may provide license request **408** (via network **30** and/or network **32**) to media distribution system **18**. Additionally, if defined within personal media device **12**, a user ID **410** (e.g., enumerating the user credentials described above) may also be included within license request **408**. As discussed above, the user credentials (i.e., included within user ID **410**) may include, but are not limited to, a user name, a user password, a user key, a device name, a device password, a device key, and/or one or more digital certificates. Prior to being provided to media distribution system **18**, DRM process **10** may digitally sign license request **408** using device private key **400**.

[**0117**] A digital signature is an electronic signature that uses the private key/public key encryption methodology (described above) and allows a sender of a message to authenticate their identity and the integrity of message sent. A digital signature may be used with both encrypted and non-encrypted messages and does not impede the ability of the receiver of the message to read the message.

[**0118**] For example, assume that DRM process **10** digitally signed license request **408** prior to providing license request **408** to media distribution system **18**. When digitally signing license request **408**, a mathematical function is typically performed on the content of license request **408**. For example, a message hash of license request **408** may be calculated by personal media device **12**, such that a message hash is the mathematical output of a known one-way hash function that transforms a string of characters (e.g., license request **408**) into a usually shorter fixed-length value that represents the original string of characters. As the hashing function is a one-way mathematical function, once a message hash is generated, the original message cannot be retrieved by processing the message hash. DRM process **10** may then encrypt the message hash (using device private key **400**) to create the digital signature (not shown). This digital signature may be attached to license request **408**. Accordingly, while the digital signature is encrypted, the original message (i.e., license request **408**) need not be. Therefore, license request **408** may be processed by media distribution system **18** even if the digital signature is not processed.

[**0119**] Continuing with the above-stated example, license request **408** and the digital signature may be received by media distribution system **18**, and media distribution system **18** may use the same hash function to generate a message hash of license request **408**. Media distribution **408** will also decrypt the digital signature received from personal media device **12** using device public key **402** (included within device digital certificate **404**) to recreate the message hash calculated by personal media device **12**. Media distribution system **18** may compare the decrypted digital signature to the message hash calculated by the media distribution system **408**. If the message hashes match, the integrity of license request **408** and the identity of personal media device **12** are both validated.

[**0120**] Additionally, the integrity of device digital certificate **404** (and, therefore, device public key **402**) may be verified when license request **408** is received from personal media device **12**. Digital certificates are typically issued and digitally signed by e.g., certification authority **412** using CA private key **414**. Accordingly, device digital certificate **404** may be verified by obtaining the CA public key **416** to verify the digital signature of device digital certificate **404**.

[**0121**] Once challenge **406**, device digital certificate **404**, and user ID **410** (i.e., license request **408**) are received by media distribution system **18**, media distribution system **18** may access data store **418** to retrieve subscription information concerning user **14** (i.e., the user defined within user ID **410**) and determine e.g., the date at which the current subscription of user **14** will expire. Data store **418** may be maintained on storage device **34** coupled to computer **28**.

[**0122**] Assume, for illustrative purposes, that media distribution system **18** is configured to automatically bill each

subscriber on the first of each month for the subscription fee for the upcoming month. Accordingly, on 01 Mar. 2005, user **14** will be billed for the cost of their March 2005 subscription. Therefore, if media distribution system **18** retrieves subscription information concerning user **14** on 06 Mar. 2005, the subscription information retrieved will indicate that user **14** has a valid subscription until 31 Mar. 2005.

[**0123**] When license request **408** is received, media distribution system **18** may retrieve subscription information concerning user **14**. In this example, the subscription information will indicate that user **14** is a valid subscriber (to media distribution system **18**) through 31 Mar. 2005.

[**0124**] Media distribution system **18** may generate a timeout indicator **420**, which indicates e.g., the user's subscription information and the expiration date of the user's current subscription. In this example, timeout indicator **420** will indicate e.g., that the subscription of user **14** will expire on 31 Mar. 2005. Media distribution system **18** obtains user encryption key **422** (i.e., the encryption key for user **14**) from data store **418**. Media distribution system **18** encrypts the user encryption key **422**, using device public key **402**, to generate encrypted user encryption key **422'** (shown with a hash fill). Timeout indicator **420**, challenge **406**, device digital certificate **404** (including device public key **402**), user ID **410**, and encrypted user encryption key **422'** may be combined (by media distribution system **18**) to form device license **424**.

[**0125**] Device license **418** may further include a system time indicator **426**, which indicates the system time as defined by media distribution system **18**. System time indicator **426** may be used to synchronize a system clock **194** (FIG. **3**) included within personal media device **12** with a system clock **428** included within media distribution system **18**.

[**0126**] Device license **424** may further include a licensing service (i.e., LS) digital certificate **430**, which typically includes a licensing service (i.e., LS) public key **432**.

[**0127**] Media distribution system **18** may digitally sign device license **424** using licensing service (i.e., LS) private key **434** (of media distribution system **18**) and provide device license **424** to personal media device **12**. Licensing system private key **434** may be stored on data store **418**.

[**0128**] When device license **424** is received from media distribution system **18**, DRM process **10** may verify the integrity of LS digital certificate **430** (and, therefore, LS public **432**). As discussed above, digital certificates are typically issued and digitally signed by e.g., certification authority **412** using CA private key **414**. Accordingly, LS digital certificate **430** may be verified by obtaining the CA public key **416** to verify the digital signature of LS digital certificate **430**.

[**0129**] DRM process **10** may use LS public key **432** (included within LS digital certificate **430**) to verify device license **424** (which was digitally signed using LS private key **434**). DRM process **10** may additionally verify challenge value **406**, device public key **402**, and the device serial number (included within device digital certificate **404**) to ensure that device license **424** is intended for personal media device **12**. DRM process **10** may then decrypt, with device private key **400**, encrypted user encryption key **422'** (that was encrypted using device public key **402**) to generate user encryption key **422** (which may be stored in non-volatile memory, examples of which include ROM **152** (FIG. **3**) and/or storage device **66** (FIG. **3**). User ID **410**, user encryption key **422**, and timeout indicator **420** may be saved on e.g., non-volatile memory, examples of which include ROM **152** (FIG. **3**) and/or storage device **66** (FIG. **3**), for use when personal media device **12** renders media content downloaded from media distribution system **18**. Additionally, as will discussed below in greater detail, DRM process **10** may retain a copy of device license **424** for use when transferring media content between personal media device **12** and e.g., personal media device **40**.

Obtaining Media Content

[**0130**] As discussed above, once user **14** subscribes to media distribution system **18**, user **14** may obtain from media distribution system **18** media content (for use with personal media device **12**) in the form of: purchased downloads received from media distribution system **18** (i.e., media content licensed to e.g., user **14** for use in perpetuity); subscription downloads received from media distribution system **18** (i.e., media content licensed to e.g., user **14** for use while a valid subscription exists with media distribution system **18**); and media content streamed from media distribution system **18**, for example.

[**0131**] Referring also to FIG. **13**, each media data file **450**, **452**, **454**, **456**, **458** downloadable from media distribution system **18** may be encrypted using a unique CEK (i.e., content encryption key) **460**, **462**, **464**, **466**, **468** respectively. For example, if media distribution system **18** includes 1,000,000 media data files available for downloading to e.g., personal media device **12**, media distribution system **18** will encrypt each media data file using a unique encryption key. Accordingly, for 1,000,000 media data files, 1,000,000 unique CEK's will be required, each of which is bound to the media data file to which the CEK is related. Accordingly, CEK **460** is bound to media data file **450**, and CEK **462** is bound to media data file **452**, for example.

[**0132**] Each CEK (e.g., keys **460**, **462**, **464**, **466**, **468**) may be a symmetric encryption key, in that the key used to encrypt a media data file may also be used to decrypt the same media data file. Additionally, each media data file may be stored on e.g., storage device **34** attached to computer **28**.

[**0133**] As discussed above, search window **304** (FIG. **10**) of proxy application **98**, may allow user **14** to search for media data files. Additionally, user **14** may download media data files from media distribution system **18** for use on personal media device **12** by selecting the download button **352** (FIG. **10**) corresponding to the media data file to be downloaded.

[**0134**] Once the download of a media data file is initiated, personal media device **12** submits the appropriate request(s) to media distribution system **18**. For example, assume that user **14** desired to download three media data files, namely media data files **450**, **454**, **456**. DRM process **10** would submit requests **470**, **472**, **474** respectively, each of which requests the desired file. For security and authentication purposes, requests **470**, **472**, **474** may be e.g., encrypted by personal media device **12** (using e.g., LS public key **432**) and/or digitally signed by personal media device **12** (using e.g., device private key **400**). Accordingly, if a request is encrypted (using e.g., LS public key **432**), the encrypted

request may subsequently be decrypted by media distribution system **18** using LS private key **434**. Further, if a request is digitally signed (using e.g., device private key **400**), the signed request may subsequently be verified by media distribution system **18** using device public key **402**.

[**0135**] Once e.g., requests **470**, **472**, **474** are received and processed by media distribution system **18**, media distribution system **18** may retrieve the requested media data files **450**, **454**, **456** from e.g., storage device **34**. As discussed above, each media data file is currently encrypted using a unique CEK, such that the CEK is bound to the media data file.

[**0136**] Prior to being downloaded to personal media device **12**, each media data file to be downloaded is bound to the user (e.g., user **14**) who requested the download. As discussed above, during device initialization, personal media device **12** provides license request **408** to media distribution system **18**. Media distribution system **18** in turn processes license **408** and obtains current subscription information concerning the user associated with license request **408** (e.g., user **14**). As discussed above, this initialization process may occur periodically and, therefore, may occur at the time that personal media device **12** is placed into docking cradle **60** (FIG. **2**). Accordingly and for this example, assume that personal media device **12** has provided the required user credentials to properly access media distribution system **18**. As discussed above, the user credentials provided to media distribution system **18** may include, but are not limited to, a user name, a user password, a user key, a device name, a device password, a device key, and/or one or more digital certificates.

[**0137**] Once media distribution system **18** retrieves the requested media data files **450**, **454**, **456** from e.g., storage device **34**, media distribution system **18** binds the retrieved media distribution files **450**, **454**, **456** to user **14** e.g., the user requesting the media data files, thus creating bound media data files **476**, **478**, **480**. Accordingly, the content encryption key (e.g., CEK **460**) associated with each media data file (e.g., media data file **450**) is encrypted using the encryption key (e.g., user encryption key **422**) of the user requesting the media data files (e.g., user **14**). Accordingly, CEK **460** is encrypted to generate CEK **460'**, CEK **464** is encrypted to generate CEK **464'**, and CEK **466** is encrypted to generate CEK **466'**. Once encrypted, bound media data files **476**, **478**, **480** (including encrypted CEK's **460'**, **464'**, **466'** respectively) are provided to personal media device **12**. As the CEK of each bound media data files **476**, **478**, **480** is encrypted using e.g., user encryption key **422**, bound media data files **476**, **478**, **480** may only be processed (e.g., rendered) by a personal media device is possession of user encryption key **422**. As discussed above, a copy of user encryption key **422** is stored on non-volatile memory within personal media device **12**. Once bound media data files **476**, **478**, **480** are received by personal media device, they may be stored on e.g., storage device **66** within personal media device **12**.

Media Content Playback

[**0138**] As discussed above, user ID **410**, user encryption key **422**, and timeout indicator **420** may be saved for use when personal media device **12** renders media content downloaded from media distribution system **18**.

[**0139**] Continuing with the above-stated example, if user **14** wishes to render one of bound media data files **476**, **478**, **480**, user **14** may select the appropriate media data file via the controls (e.g., backward skip switch **78**; forward skip switch **80**; play/pause switch **82**; menu switch **84**; radio switch **86**; and slider assembly **88**, for example) and display panel **90** of personal media device **12**. Once one or more media data files are selected for playback, the appropriate file(s) are retrieved from e.g., storage device **66**. As discussed above, prior to each media data file being provided to personal media device **12**, the CEK of each media data file may be encrypted (by media distribution system **18**) using user encryption key **422**. As discussed above, user encryption key **422** may be a symmetric encryption key and, therefore, the key used to e.g., encrypt CEK **460** may also be used to decrypt encrypted CEK **460'**.

[**0140**] Once the appropriate bound media data files are retrieved from e.g., storage device **66**, DRM process **10** may decrypt the appropriate CEK (using user encryption key **422**) so that the media data file can be processed and rendered on personal media device **12**. For example, if user **14** wished to render bound media data files **476**, **478**, personal media device **12** would decrypt encrypted CEK **460'** to generate CEK **460**. CEK **460** may then be used by DRM process **10** to decrypt media data file **450** for playback by personal media device **12**. Further, DRM process **10** would decrypt encrypted CEK **464'** to generate CEK **464**. CEK **464** may then be used by DRM process **10** to decrypt media data file **454** for playback by personal media device **12**.

[**0141**] Typically, prior to processing and rendering e.g., bound media data files **476**, **478**, DRM process **10** will verify that e.g., user **14** has sufficient rights to process and render the bound media data files.

[**0142**] As discussed above, media distribution system **18** is typically a subscription-based service, in that e.g., user **14** subscribes to media distribution system **18** and pays e.g., a monthly subscription fee to be granted access to media distribution system **18**. Further, user **14** may obtain from media distribution system **18** subscription downloads that allow user **14** to process and playback the subscription downloads only while a valid subscription exists with media distribution system **18**.

[**0143**] Assuming that bound media data files **476**, **478**, **480** are subscription downloads (as opposed to purchased downloads that are licensed in perpetuity for use by user **14**), prior to rendering and/or processing bound media data files **476**, **478**, **480**, DRM process **10** may obtain timeout indicator **420**, which as discussed above may be stored on e.g., non-volatile memory, examples of which include ROM **152** (FIG. **3**) and/or storage device **66** (FIG. **3**). DRM process **10** may then compare the expiration date (e.g., **31** March **2005**) defined within timeout indicator **420** to the date and/or time defined within system clock **194** to determine if e.g., user **14** is still allowed to render bound media data files **476**, **478**, **480**. In this example, as user **14** has a valid subscription through **31** March **2005** and the current date and time (as defined by system clock **194**) is 17:53 GMT on 06 Mar. 2005, the subscription of user **14** (with respect to media distribution system **18**) is valid and current. Accordingly, bound media data files **476**, **478**, **480** may be processed for playback.

Device-to-Device Media Content Transfer

[**0144**] As discussed above, media distribution system **18** is typically a subscription-based service, in that e.g., user **14**

subscribes to media distribution system **18** and pays e.g., a monthly subscription fee to be granted access to media distribution system **18**. Further, user **14** may obtain from media distribution system **18** subscription downloads that allow user **14** to process and playback the subscription downloads only while a valid subscription exists with media distribution system **18**. Accordingly, since the rights associated with a subscription download are based upon the existence of a valid subscription with media distribution system **18**, subscription downloads may be transferred from a first personal media device to a second media device, as long as a valid subscription exists concerning the second personal media device.

[**0145**] Referring also to FIG. **14** and continuing with the above-stated example, assume that user **14** has downloaded bound media data files **476**, **478**, **480** which are stored on e.g., storage device **66** within personal media device **12**. Further, assume that user **26** desired to obtain a copy of bound media data file **476** for playback on personal media device **40**. As discussed above, when a device is initialized, a copy of a device license is transferred to and retained on the personal media device for use when transferring media content between personal media devices. Accordingly, personal media device **12** includes source device license **424** and personal media device **40** includes target device license **500**.

[**0146**] Typically, a device-to-device content transfer is initiated by the user of the source device. In the above-stated example, personal media device **12** is the source device and personal media device **40** is the target device. Accordingly, user **14** may initiate the transfer of bound media data file **476** from personal media device **12** to personal media device **40**.

[**0147**] Referring back to FIG. **2**, if e.g., user **14** wishes to transfer a media data file to another personal media device, user **14** may depress menu switch **84**, resulting in the generation of e.g., pop-up menu **106**. Using slider assembly **88**, user **14** may select the "Share Content" command **108** from pop-up menu **106**, resulting in the generation of content window **110**. From content window **110**, user **14** may select the appropriate file for transfer. Assume that user **14** selects "Peggy Sue", which corresponds to bound media data file **476**. Once user **14** selects the track for transfer, device application **64** may render a transfer window **112** that e.g., includes a track title field **114** that identifies the title of the track being transferred and an artist field **116** that identifies the artist of the track being transferred.

[**0148**] Transfer window **112** may include a transfer button **118** (selectable via slider assembly **88**) for initiating the transfer of bound media data file **476** to e.g., personal media device **40**. In this example, if user **14** selects transfer button **118** with slider assembly **88**, the transfer of bound media data file **476** (i.e., "Peggy Sue" from "Buddy Holly") from personal media device **12** to (in this example) personal media device **40** is initiated. Transfer window **112** may include a transfer status indicator **120** for indicating the progress of the transfer of e.g., "Peggy Sue" by "Buddy Holly". Transfer window **112** may further include a cancel button **122** for allowing user **14** to cancel the file transfer and close download window **112**.

[**0149**] Referring now to FIG. **14**, once the transfer of bound media data file **476** is initiated, the devices may exchange device digital certificates for authentication purposes. For example, DRM process **10** may provide source device digital certificate **404** (which includes source device public key **402**) to device personal media device **40** for authentication. As discussed above, the integrity of source device digital certificate **404** (and, therefore, source device public key **402**) may be verified (by personal media device **40**) via CA public key **416** (a copy of which is typically stored in non-volatile memory **502** of personal media device **40**), as source device digital certificate **404** was issued and digitally signed by e.g., certification authority **412** (FIG. **12**) using CA private key **414** (FIG. **12**).

[**0150**] Further, personal media device **40** may provide target device digital certificate **504** (which includes target device public key **506**) to device personal media device **12** for authentication. The integrity of target device digital certificate **504** (and, therefore, target device public key **506**) may be verified by DRM process **10** via CA public key **416** (a copy of which is typically stored in non-volatile memory **66**/**152** of personal media device **12**), as target device digital certificate **504** would typically also have been issued and digitally signed by e.g., certification authority **412** (FIG. **12**) using CA private key **414** (FIG. **12**).

[**0151**] As shown in FIG. **3**, personal media devices (e.g., personal media device **12**) may include a wireless interface **182** for wirelessly-coupling personal media device **12** to network **30** (or network **32**) and/or other personal media devices. Wireless interface **182** may be coupled to an antenna assembly **184** for RF communication to e.g., WAP **52**, and/or an IR (i.e., infrared) communication assembly **186** for infrared communication with e.g., a second personal media device (such as personal media device **40**). Accordingly, communication between personal media devices **12**, **40** may occur wirelessly via RF communication and/or infrared communication. Additionally, an external connector (not shown) may be included within each personal media device that allows for the hardwired-interconnection of multiple personal media devices.

[**0152**] Once certificates **404** and **504** are verified, personal media device **40** provides target device license **500** to personal media device **12**. As with device license **424** (FIG. **11**), target device license **500** may include: LS digital certificate **508** (which includes LS public key **432**), system time indicator **512**, timeout indicator **514** (i.e., for the subscription of user **26**), encrypted user encryption key **516** (i.e., for user **26**), user ID **518** (i.e., for user **26**), challenge **520**, and target device digital certificate **502** (which includes a copy of target device public key **504**).

[**0153**] Upon receiving target device license **500** from personal media device **40**, DRM process **10** may verify the integrity of target device license **500**. Accordingly, DRM process **10** may verify the integrity of LS digital certificate **508** (and, therefore, LS public key **432**). As discussed above, digital certificates are typically issued and digitally signed by e.g., certification authority **412** (FIG. **12**) using CA private key **414** (FIG. **12**). Accordingly, LS digital certificate **508** may be verified by DRM process **10** using CA public key **416**.

[**0154**] DRM process **10** may use LS public key **432** (included within LS digital certificate **508**) to verify target device license **500** (which was digitally signed using LS private key **434** (FIG. **12**)). DRM process **10** may additionally verify that user **26** has a valid subscription to media

distribution system **18** by comparing timeout indicator **514** to system clock **194**. For example, as user **26** has a valid subscription through 22 Mar. 2005 (as defined by timeout indicator **514**) and the current date and time (as defined by system clock **194**) is 22:06 GMT on 13 Mar. 2005, the subscription of user **26** (with respect to media distribution system **18**) is valid and current.

[**0155**] Assuming that the integrity of target device license **500** is verified, the transfer of bound media data file **476** may begin. Depending on the manner in which DRM system **10** is configured, user **26** may be required to have a valid and current subscription (with media distribution system **18**) prior to initiating the transfer of any media data files to personal media device **40**. However and as discussed above, since the personal media device checks for the existence of a valid and current subscription prior to rendering media data files, even is the transfer was effectuated while user **26** did not have a valid and current subscription with media distribution system **18**, user **26** would be prohibited from rendering the transferred media data files.

[**0156**] In order to effectuate the media data file transfer, DRM process **10** generates a random session key (i.e., RSK) **522**, which is encrypted using target device public key **504** (included within target device digital certificate **504**) to generate encrypted RSK **522'**. DRM process **10** provides encrypted RSK **522'** to personal media device **40**, which is decrypted (using target device private key (not shown)) to retrieve RSK **522**. RSK **522** may be a 1024-bit symmetric encryption key.

[**0157**] As personal media device **12** and personal media device **40** each contain a copy of RSK **522**, a secure communication channel **524** may be established between devices **12**, **40**, in which all data transmitted across secure communication channel **524** is encrypted (using RSK **522**) prior to transmission and decrypted (using RSK **522**) upon receipt. Secure communication channel **524** may be a wireless communication channel (using e.g., RF communication and/or infrared communication), or a wired communication channel (using an external connector (not shown) on devices **12**, **40**).

[**0158**] DRM process **10** may retrieve (from e.g., storage device **66**) bound media data file **476** for transmission to personal media device **40**. However and as discussed above, as CEK **460'** of bound media data file **476** was encrypted using the encryption key of user **12** (e.g., user encryption key **422**), bound media data file **476** will not be accessible (in its current form) by user **26**. Therefore, bound media data file **476** must be unbound from user **12** and bound to user **26**. Accordingly, DRM process **10** obtains bound media data file **476** from e.g., storage device **66** and decrypts CEK **460'** (using user encryption key **422**) to obtain CEK **460**. Unbound media data file **526** may be transferred (via secure communication channel **524**) from personal media device **12** to personal media **40**. Upon receipt, personal media device **40** may encrypt CEK **460** of unbound media data file **526**, using the encryption key of user **26** (i.e., user encryption key **528**) to generate bound media data file **530**, which includes encrypted CEK **460"**. Personal media device **40** may store bound media data file **530** for subsequent rendering in non-volatile memory **502**.

[**0159**] User encryption key **422** is described above as typically being a symmetric encryption key, in that the same key that is used to encrypt a CEK may also be used to decrypt the encrypted version of the CEK. Further and as described above, the same user encryption key **422** is used to encrypt all CEK's. Therefore, if one-hundred bound media data files are downloaded to and stored upon personal media device **12**, the same user encryption key **422** may be used to decrypt each of the one-hundred encrypted CEKs. However, other configurations of user encryption key **422** are possible. For example, user encryption key **422** may be a symmetric key block, as opposed to a single symmetric key.

[**0160**] Referring now to FIG. **15**, a 32-byte (i.e., 256-bit) symmetric key block **600** is shown. Assume for this example that a 16-byte (i.e., 128-bit) key is used to encrypt and decrypt each encrypted CEK. Through the use of one e.g., 256-bit symmetric key block **600**, multiple 128-bit symmetric keys (e.g., user encryption keys **602**, **604**, **606**, **608** may be defined. For example, a first user encryption key **602** may be defined as bits **000-127** of symmetric key block **600**. A second user encryption key **604** may be defined as bits **004-131** of symmetric key block **600**. A third user encryption key **606** may be defined as bits **128-255** of symmetric key block **600**. A fourth user encryption key **608** may be defined as bits **124-251** of symmetric key block **600**. Accordingly, a plurality of unique symmetric user encryption keys may be defined using a single symmetric key block **600**. Accordingly, to properly define the individual user encryption keys, in this particular example a bit shift parameter **610** is defined for each user encryption key **602**, **604**, **606**, **608**, which defines the starting point of the respective key. For example, user encryption key **602** starts at bit-**0** of symmetric key block **600** and, therefore, has a bit shift **610** of 0-bits. As user encryption key **604** starts at bit-**4** of symmetric key block **600**, user encryption key **604** has a bit shift **610** of 4-bits. As user encryption key **606** starts at bit-**128** of symmetric key block **600**, user encryption key **606** has a bit shift **610** of 128-bits. As user encryption key **608** starts at bit-**124** of symmetric key block **600**, user encryption key **608** has a bit shift **610** of 124-bits.

[**0161**] While various user encryption keys are defined within symmetric key block **600** by shifting the starting point of each individual user encryption key, other configurations are possible. For example, keys may be defined using only odd or even bits in conjunction with a bit shift. Additionally and/or alternatively, keys may be defined within symmetric key block **600** algorithmically, in that an algorithm is used to define the individual bits used (within symmetric key block **600**) to define a unique user encryption key.

Systems and Methods of Using Personal Media Device

[**0162**] Various systems and methods of using a personal media device are described below. Each of these systems and methods may be implemented on a personal media device **12** and in connection with a media distribution system **18**, for example, as described above. The systems and methods may be implemented using one or more processes executed by personal media device **12**, proxy computer **54** and/or server computer **28**, for example, in the form of software, hardware, firmware or a combination thereof. Each of these systems and methods may be implemented independently of the other systems and methods described herein. As described above, personal media device

12 may include a dedicated personal media device (e.g., an MP3 player), a personal digital assistant (PDA), a cellular telephone, or other portable electronic device capable of rendering digital media data.

Subscription-Based Digital Rights Management

[0163] Referring to FIGS. 16 and 17, a system and method for providing subscription-based digital rights management (DRM) on personal media device 12 are shown. According to subscription-based DRM, personal media device 12 is licensed or registered to a user and content rights associated with licensed media content 1102 are bound to the user of personal media device 12 under a subscription to a media distribution service. Thus, a user is able to obtain and/or render licensed media content 1102 on personal media device 12 provided that personal media device 12 maintains a valid license under a user subscription.

[0164] Licensed media content 1102 may be provided as one or more media data files pre-loaded on, downloaded to, or transferred to personal media device 12. As described above, licensed media content 1102 may include media data files such as audio files (e.g., music), video files, audio/video files, and multimedia content. Licensed media content 1102 may be arranged and presented as individual media content items (e.g., musical tracks) that may be individually and selectively rendered (e.g., subscription content or purchased content) or as multiple content items (e.g., a series of musical tracks) that may only be played in a defined sequence in compliance with performance complement requirements and with limited or no user interaction (e.g., non-interactive content).

[0165] Media content 1102 licensed by media distribution system 18 may have different content rights associated with different media data files. Content rights may include, for example: non-subscription rights (e.g., purchased content rights) that allow content to be rendered and/or copied (e.g., burned to CD) by the licensed user indefinitely independent of a valid subscription; and subscription-based rights that allow content to be streamed, downloaded, rendered and/or transferred by the licensed user for a limited period of time under a valid subscription. Content rights associated with a particular media data file may be defined in a content license associated with the media data file (e.g., embedded in the media data file or in a separate content license database).

[0166] To protect licensed media content 1102, media data files may be encrypted with associated content encryption keys 1104. To bind the licensed content 1102 to a user, content encryption keys 1104 may be encrypted with user encryption key 422, for example, as described above. User encryption key 422 may be included in device license 424 with other "user credentials" such as user ID 10 and expiration data 420 (e.g., a timeout indicator) used to verify the user's subscription.

[0167] DRM process 1110 may be resident on and executed by personal media device 12 to handle digital rights management, for example, as described above. Content playback engine 1120 may be resident on and executed by personal media device 12 to perform the core functions or processes associated with rendering media content such as processing media data files. Although content playback engine 1120 and DRM process 1110 are shown as separate functional components, DRM process 1110 may be incorporated with content playback engine 1120. DRM process 1110 and content playback engine 1120 may be components of device application 64 (see FIG. 1), for example, as an embedded feature, software plug-in, or stand-alone application. The instruction sets and subroutines of DRM process 1110 and content playback engine 1120 may be executed by one or more processors (not shown) and one or more memory architectures (not shown) incorporated into personal media device 12.

[0168] An exemplary method of subscription-based DRM is illustrated in FIG. 17 and is described below. A user may be registered 1150 with media distribution system 18 to obtain a subscription to a media distribution service, for example, during the device initialization process described above. The user may also register 1150 to obtain the subscription before the device initialization process. Personal media device 12 may be registered 1152 with media distribution system 18 under the user subscription, for example, during the device initialization process described above. DRM process 1110 on personal media device 12, for example, may handle the user registration and/or device registration during device initialization. As a result of a user registration 1150 and device registration 1152, device license 424 is generated and stored on personal media device 12.

[0169] During use, personal media device 12 may receive 1154 a request by the user to obtain and/or render licensed media content 1102 licensed by media distribution system 18. Upon receiving a request to obtain and/or render a content item, personal media device 12 may determine 1160 if the content rights associated with the corresponding media data file are non-subscription rights or subscription rights. DRM process 1110 or content playback engine 1120, for example, may access the content license associated with the media data file. If the content rights are non-subscription rights, personal media device 12 may obtain and/or render 1162 the media content regardless of the subscription status. DRM process 1110 and/or content playback engine 1120, for example, may retrieve, decrypt and process the media data file associated with the selected non-subscription content item without having to verify device license 424.

[0170] If the content rights are subscription rights, personal media device 12 may verify 1164 the subscription associated with the personal media device 12. DRM process 1110 on personal media device 12, for example, may access device license 424 to determine if the subscription is valid and may compare expiration data 420 to system clock 194 to determine if the content has expired. If the subscription cannot be verified (e.g., the content expired or the subscription is invalid or never existed), personal media device 12 may attempt to renew 1166 an existing subscription or initiate a new subscription. If the subscription can be verified, personal media device 12 may obtain and/or render 1162 the licensed media content item. DRM process 1110 and/or content playback engine 1120 on personal media device 12, for example, may retrieve, decrypt and process the media data file associated with the selected subscription content item.

[0171] Accordingly, a subscription-based DRM system and method allows a user to render media content under a user subscription on a licensed personal media device 12 without having to track content licenses associated with each media data file individually.

Bulk Licensing Pre-Loaded Media Content

[0172] Referring to FIGS. **18-20**, there is shown a system and method for bulk licensing pre-loaded media content on a personal media device **12**. Personal media device **12** may be pre-loaded with media content **1200** (e.g., on storage device **66** shown in FIG. **3**), which may be licensed in bulk, for example, during device initialization. In general, pre-loaded content **1200** may be any content that is not downloaded to personal media device **12**, for example, from media distribution system **18**. Pre-loaded content **1200** may be pre-loaded by storing the content when personal media device **12** is manufactured, or may be pre-loaded by transferring the content from another storage medium provided with personal media device **12** (e.g., from a CD or DVD).

[0173] As described above, pre-loaded media content **1200** may include media data files such as audio files (e.g., music), video files, audio/video files, and multimedia content. Pre-loaded content **1200** may be arranged and presented as individual media content items (e.g., musical tracks) that may be individually and selectively rendered (e.g., subscription content) and/or as multiple content items that may only be rendered in a defined sequence in compliance with performance complement requirements and with limited or no user interaction (e.g., non-interactive content). Non-interactive content (also referred to as radio content), for example, may allow a user to start and stop rendering the sequence of content items and to skip a limited number of content items.

[0174] In one example, personal media device **12** may include about 5 to 10 gigabytes of pre-loaded content **1200** including content data for about 10 to 15 non-interactive content sequences (e.g., radio stations). Pre-loaded content **1200** may correspond to a particular type or category of media content to provide a "specialized" personal media device **12**. For example, personal media device **12** may be pre-loaded with music of a particular genre, for example, to provide a Jazz personal media device or with music of a particular artist, for example, to provide an Elvis personal media device.

[0175] According to one embodiment, personal media device **12** renders media content if personal media device **12** is initialized and registered or licensed to a user, for example, under a user subscription to a media distribution service. Pre-loaded content **1200** may be pre-loaded before personal media device **12** is initialized and licensed to a user. In contrast, downloaded content **1202** may be downloaded to personal media device **12**, for example, from media distribution system **18**, after the user initializes and licenses the device. Thus, pre-loaded content **1200** may advantageously save download bandwidth. The personal media device **12** may also enable a limited time trial of pre-loaded content **1200** without having to subscribe to a media distribution service and download content **1202**.

[0176] DRM process **1210** may be resident on and executed by personal media device **12** to handle digital rights management, for example, as described above. Content playback engine **1220** may be resident on and executed by personal media device **12** to perform the core functions or processes associated with rendering media content such as processing media data files. Although content playback engine **1220** and DRM process **1210** are shown as separate functional components, DRM process **1210** may be incorporated with content playback engine **1220**. DRM process **1210** and content playback engine **1220** may be components of device application **64** (see FIG. **1**), for example, as an embedded feature, software plug-in, or stand-alone application. The instruction sets and subroutines of DRM process **1210** and content playback engine **1220** may be executed by one or more processors (not shown) and one or more memory architectures (not shown) incorporated into personal media device **12**.

[0177] To prevent media content on personal media device **12** from being accessed or rendered without authorization, media content may be encrypted using content encryption keys (CEKs), as described above. Downloaded content **1202** may include embedded content licenses including CEKs **1208**, which may used to decrypt the downloaded content **1202**. Pre-loaded content **1200**, on the other hand, may be pre-loaded without embedded content licenses and CEKs. Thus, the user initially may not render pre-loaded content **1200** on personal media device **12** because pre-loaded content **1200** is encrypted and the CEKs for decrypting the pre-loaded content **1200** are not pre-loaded. Content licenses **1204** including CEKs **1208'** corresponding to the pre-loaded content **1210** may be downloaded in bulk, for example, when the personal media device **12** is activated. To bind media content to a subscriber or licensed user of personal media device **12**, CEKs **1208**, **1208'** may be encrypted with user encryption key **422**, for example, as described above. User encryption key **422** may then be used by the registered user or subscriber to decrypt CEKs **1208**, **1208'** needed to decrypt media content.

[0178] Individual media content items may be identified using content item identifiers, such as digital rights management (DRM) IDs used to uniquely identify content within media distribution system **18**. In one embodiment, content licenses **1204** may be provided as a sorted or indexed database with content licenses **1204** indexed using content item identifiers corresponding to content item identifiers associated with content items in pre-loaded content **1200**.

[0179] An exemplary method of bulk licensing pre-loaded content is illustrated in FIG. **19** and is described below. Personal media device **12** with pre-loaded content **1200** may establish communication with media distribution system **18**, for example, to register and license personal media device **12** during the device initialization process described above. When communication is established, media distribution system **18** may receive **1250** a request to register and license personal media device **12** for use with media distribution system **18**. Media distribution system **18** may then establish **1252** user encryption key **422** (in addition to other user credentials), for example, as described above.

[0180] As part of the process of registering and/or licensing personal media device **12**, media distribution system **18** may receive **1254** an identification of pre-loaded content **1200** to be licensed. Personal media device **12**, for example, may transmit to media distribution system **18** the content item identifiers corresponding to content items in pre-loaded content **1200**. Media distribution system **18** may obtain **1256** corresponding content licenses for pre-loaded content on personal media device **12**. Media distribution system **18**, for example, may generate a sorted or indexed pre-build database including content licenses **1204** (and CEKs **1208'**)

associated with the content item identifiers provided by personal media device **12**. Media distribution system **18** may encrypt CEKs **1208'** in the content licenses **1204** with the user encryption key **422**, for example, in the manner described above for encrypting CEKs embedded in downloaded content. Media distribution system **18** may then transmit **1260** to personal media device **12** the content licenses **1204** with encrypted CEKs **1208'** for pre-loaded content **1200**, which have been bound to user of personal media device **12**.

[**0181**] An exemplary method of rendering pre-loaded content is illustrated in FIG. **20** and described below. When a request to render one or more media content items is received **1270**, content playback engine **1220** may first attempt to access **1272** an embedded license, for example, in the header of the requested media data file. If the requested media data file includes an embedded license, a CEK in the embedded license may be decrypted **1274** with user encryption key **422**. If the requested media data file does not include an embedded license, personal media device **12** may locate **1276** the CEK associated with the requested media data file and decrypt **1280** the associated CEK with user encryption key **422**. DRM process **1210** and/or content playback engine **1220**, for example, may use the content item identifier associated with the requested content item to locate the content license (and encrypted CEK) in a database of content licenses. Once the corresponding CEK is located and decrypted, the media content item may be decrypted **1282** with the CEK and the media data file may be rendered.

[**0182**] Accordingly, the system and method of bulk licensing enables pre-loaded content to be securely provided on a personal media device and then licensed to a user of the personal media device at a later time in an efficient manner.

Accessing Pre-Loaded Media Content

[**0183**] Referring to FIGS. **21-25**, a system and method for loading content onto a personal media device **12** and providing access to the content upon the sale of the personal media device are shown. Personal media device **12** may be pre-loaded with media content **1300** (e.g., on storage device **66** shown in FIG. **3**) from a Media Content Pre-Load System **1302**. In general, the pre-loaded content may be any content that is loaded on personal media device **12** prior to the device being sold. For example, the pre-loaded media content may be pre-loaded by storing the content when personal media device **12** is manufactured or at any point prior placing the device up for sale (e.g., displaying the device on a store self).

[**0184**] As described above, pre-loaded media content **1300** may include media data files such as audio files (e.g., music), video files, audio/video files, multimedia content or other similar type of data. Pre-loaded content **1300** may be arranged and presented as individual media content items (e.g., musical tracks) that may be individually and selectively rendered (e.g., subscription content) and/or as multiple content items that may only be rendered in a defined sequence in compliance with performance complement requirements and with limited or no user interaction (e.g., non-interactive content). Non-interactive content (also referred to as radio content), for example, may allow a user to start and stop rendering the sequence of content items and to skip a limited number of content items.

[**0185**] In one example, personal media device **12** may include about 5 to 10 gigabytes of pre-loaded content **1300** including content data for about 10 to 15 non-interactive content sequences (e.g., radio stations). Pre-loaded content **1300** may correspond to a particular type or category of media content to provide a "specialized" personal media device **12**. For example, personal media device **12** may be pre-loaded with music of a particular genre, for example, to provide a Jazz personal media device or with music of a particular artist, for example, to provide an Elvis personal media device or with music from a particular time period, for example, to provide a 1960's personal media device or other type of genre or combination of genres.

[**0186**] According to one embodiment, to provide a particular genre, a group of media files **1304** (e.g., MP3 files) is pre-loaded from Media Content Pre-load System **1302**. For example, for a 1960's personal media device, each media file included in group **1304** may contain a unique musical piece from the 1960's decade. By pre-loading content onto personal media device **12**, a purchaser of the device does not need to expend time to download media content before using the device. Also, by providing pre-loaded content, a purchaser may begin to use personal media device **12** as soon as the device is purchased. So, using this example, a purchaser may listen to a collection of 1960's music as soon as he or she removes the device from the selling container. Furthermore, the purchaser may use personal media device **12** prior to entering into a subscription service or other type of agreement.

[**0187**] To prevent media content on personal media device **12** from being accessed or rendered without authorization, pre-loaded content **1300** may be encrypted using a single encryption key, referred to as a set encryption key (SEK) **1306**. Similar to pre-loaded content **1300**, SEK **1306** is pre-loaded onto personal media device **12** prior to placing the device up of sale. SEK **1306** may be stored in a secure portion of memory (referred to as a key store **1308**) that is included in personal media device **12**. SEK **1306** may implement one or more security techniques such as techniques associated with the Helix Security Manager produced by RealNetworks, Inc. of Seattle, Wash.

[**0188**] By using a single encryption key (i.e., SEK **1306**) for an entire set of pre-loaded content (e.g., media file group **1304**), less memory capacity and downloading time is needed in comparison to assigning a unique encryption key to each individual media file included in the pre-loaded content. A unique SEK may also be assigned to each different set of pre-loaded content, if multiple sets of content are pre-loaded. For example, one unique SEK may be stored on each personal media device that is pre-loaded with a set of 1960's music while another SEK may be stored on each personal media device that is pre-loaded with a set of Elvis Presley music.

[**0189**] SEK **1306** provides access to the pre-loaded content as soon as a purchaser powers up personal media device **12**. In some embodiments, personal media device **12** may be offered for sale (e.g., placed on a store shelf) with partially charged batteries so that a purchaser may use the device upon purchase. Upon being powered up, SEK **1306** may be used to decrypt the pre-loaded content **1304** for rendering to the purchaser. While this embodiment describes pre-loading one group of content **1304** and one corresponding SEK, in some embodiments, multiple sets of content (e.g., the 1960's music set, the 1970's music set, the 1980's music set, etc.)

may be pre-loaded onto personal media device **12**. Correspondingly, a unique SEK for each content set is pre-loaded onto personal media device **12** so that the respective content in each set may be decrypted and rendered.

[**0190**] Along with SEK **1306**, SEK expiration data **1310** is pre-loaded onto personal media device **12** for placing limits on the user's accessing of pre-loaded content **1300**. By limiting access, an incentive is provided for the user to enter into a subscription to continue to access the pre-loaded content **1300** and for downloading additional media content. SEK expiration data **1310** may implement one or more techniques to limit access to pre-loaded content **1300**. For example, SEK expiration data may allow access to the pre-loaded content **1300** for a predefined period of time. Using an internal clock (such as internal clock **194** shown in FIG. **3**), access may be granted for a period of time (e.g., 4 months) after the pre-loaded content is stored on the personal media device **12**. In another embodiment, access to the pre-loaded content **1300** may be limited for a period of time after the content is initially accessed. For example, access to pre-loaded content **1300** may be granted for thirty days after the first instance that the content (e.g., media file **2**) is accessed. In still another exemplary embodiment, the number of access instances may be limited. For example, after a predefined number of access instances has occurred, access to the pre-loaded content **1300** may be denied. To track access instances, a counter **1311** included in SEK expiration data **1310** may be preset for a particular value (e.g., **1000**) by media content pre-load system **1302**. As each access instance occurs, the counter **1311** may be decremented (e.g., by 1). Upon being decremented to a zero value, access to the pre-loaded content **1300** may be blocked. However, as described below, if the user enters into a subscription, pre-loaded content **1300** may be accessed even if the value being held in counter **1311** has decremented to a zero value.

[**0191**] In some embodiments, the predefined value (referred to as a playcount) assigned to the counter **1311** may be determined from statistics associated with typical content access. For example, based upon a typical content access rate of five hundred accesses instances per month, by providing a playcount of one thousand access instances, a user may be provided access to pre-loaded content **1300** equivalent to a two-month trial period. As described below, along with internally tracking the decrementing of the playcount on personal media device **12**, tracking may also be externally monitored. For example, by tracking each instance that a particular pre-loaded media file is accessed, a count may be computed and provided to a particular music entity (e.g., a record label) associated with the accessed media file.

[**0192**] DRM process **1210** may be resident on and executed by personal media device **12** to handle digital rights management, for example, as described above. Content playback engine **1220** may be resident on and executed by personal media device **12** to perform the core functions or processes associated with rendering media content such as processing the pre-loaded media files. Although content playback engine **1220** and DRM process **1210** are shown as separate functional components, DRM process **1210** may be incorporated with content playback engine **1220**. DRM process **1210** and content playback engine **1220** may be components of device application **64** (see FIG. **1**), for example, as an embedded feature, software plug-in, or stand-alone application. The instruction sets and subroutines of DRM process **1210** and content playback engine **1220** may be executed by one or more processors (not shown) and one or more memory architectures (not shown) incorporated into personal media device **12**.

[**0193**] An exemplary method of pre-loading content is illustrated in FIG. **23** and is described below. Personal media device **12** may establish communication with media content pre-load system **1302**, for example, to download a collection of media content associated with a particular genre during the pre-load process described above. When communication is established, media content pre-load system **1302** may provide **1400** pre-load content to the personal media device **12**. Additionally, once communication is established, media content pre-load system **1302** may provide **1402** a SEK to the personal media device. As described above, the SEK is uniquely associated with the collection of pre-loaded content and is used to decrypt each media file included in the content collection. Also while communication is established, media content pre-load system **1302** may provide **1404** SEK expiration data to the personal media device. As described above, a portion of this data may represent an expiration date, an expiration period and/or an initial playcount (that may be decremented for each access instance).

[**0194**] An exemplary method of determining if the pre-loaded may be accessed is illustrated in FIG. **24** and is described below. Based upon user interaction, the personal media device **12** may access **1406** of the pre-loaded SEK (e.g., SEK **1306**). Once accessed, the personal media device **12** may determine **1408** if the SEK is valid. For example, the SEK may be valid if an expiration date has not occurred or if a playcount has not been decremented to a value of zero. If the SEK is valid, the personal media device **12** may provide **1410** access to the pre-loaded content. In some embodiments, the personal media device **12** may also adjust data associated with the SEK. For example, the playcount associated with the SEK may be decremented. If the SEK is not valid, the personal media device may deny **1412** access to the pre-loaded content.

[**0195**] Referring to FIG. **22**, there is shown a system for loading additional content onto a personal media device **12** that is already storing pre-loaded content (e.g., pre-load content **1300**). Based upon SEK expiration data **1310**, pre-loaded content **1300** may be accessible for rending on the personal media device **12**. For example, if a playcount (stored in the counter **1311**) has a nonzero value, access to pre-loaded content **1300** is granted. However, if the SEK **1308** is not valid, access to pre-loaded content **1300** is typically denied.

[**0196**] By registering the personal media device **12** and entering into an agreement (e.g., a subscription) with media distribution system **18**, access to pre-load content **1300** may be re-stored. In particular, by purchasing a subscription, SEK **1306** is considered valid and may be used to decrypt the pre-loaded content **1300**.

[**0197**] Along with re-establishing access to the pre-loaded content **1300**, a valid subscription allows the personal media device **12** to download additional content from the media distribution system **18**. In an exemplary embodiment, personal media device **12** is communicates with media distribution system **18** via a connection to a computer system or by another technique described above. Upon establishing

communication with media distribution system **18**, a user may be queried if he or she would like to register the personal media device **12** and obtain a subscription service (or other type of valid license). If a subscription is requested and the personal media device **12** is registered with media distribution system **18**, a user encryption key **1500** is provided to the device. In this embodiment, user encryption key **1500** is securely stored in key store **1308**, however, in other embodiments, the user encryption key may be stored at one or more other location in personal media device **12**.

[**0198**] User encryption key **1500** is used to decrypt media content (i.e., non pre-loaded media content) that is downloaded from media distribution system **18**. For example, a user may desire to download media content **1502**, which is different from the pre-loaded media content **1300**. By using the user encryption key **1500**, each of the media files **1504** included in the newly downloaded content **1502** may be decrypted and rendered on personal media device **12**. As described above, various types of content may be downloaded such as purchased media, subscription media, and other similar types of content.

[**0199**] Along with the user encryption key **1500**, subscription data **1506** may also be downloaded from the media distribution system **18** during the registration and subscription process. Similar to the set encryption key expiration data **1310** (in regards to the set encryption key **1306**), the subscription data **1506** is used to determine if the user encryption key **1500** is valid. For example, subscription data **1506** may represent a time period or particular date to which the subscription is valid. Thereby, if the time period has not expired or the particular date has not been reached, the user encryption key **1500** may be used to decrypt downloaded content **1502**. If the user encryption key **1500** is not valid (e.g., a time period represented in subscription data **1506** has expired), the personal media device **12** is denied access to the downloaded content **1502**. To re-establish the validity of user encryption key **1500**, the subscription data **1506** may be updated by establishing communication between personal media device **12** and media distribution system **18**. Once the subscription data **1506** is updated, the downloaded content **1502** may be accessed (for the new subscription period). Additionally, once the subscription data **1506** is updated, the SEK **1306** may once again be used to access the pre-loaded content **1300**. If the subscription data **1506** is not updated, the user encryption key **1500** may not be used to decrypt the downloaded content **1502** and future downloading from media distribution system is denied. In this scenario, accessing the pre-loaded content **1300** is dependent upon whether the SEK **1306** is still valid.

[**0200**] An exemplary method for downloading and accessing additional content on a personal media device is illustrated in FIG. **25** and is described below. The personal media device **12** may establish **1600** a communication link with the media distribution system **18**. Once communication is established, information may be provided **1602** from the personal media device **12** to the media distribution system **18**. For example, data representative of the pre-loaded content present on the personal media device **12** may be provided to the media distribution system **18**. By knowing which set of pre-loaded content is present on the device, other sets of media content or individual media content files may be offered for downloading onto the personal media device **12**. Data representative of the SEK expiration data **1310** and/or the subscription data **1506** may be provided to the media distribution system **18**. The SEK expiration data **1310** may be monitored by the media distribution system **18**, for example, to determine if the SEK **1306** is valid. The SEK expiration data **1310** may also be metered, for example, to determine the current value of a playcount, which individual media files have been accessed, and how many access instances have occurred for each of the individual media files.

[**0201**] Once the information is provided, the media distribution system **18** may determine **1604** if a user encryption key is present on the personal media device and if the user encryption key is valid. If no key is present or the user encryption key is not valid (e.g., subscription data **1506** indicates the user's subscription has expired), the user may be queried **1606** if he or she would like to attain a valid user encryption key. If the user is not interested, access is denied **1608** to downloading additional media content from the media distribution system **18**. If the user indicates that he or she would like a subscription, the user may be lead through a subscription registration process **1610** (e.g., enter into a subscription agreement) to attain a valid user encryption key.

[**0202**] If a valid user encryption key is present on the personal media device **12**, or once a valid user encryption key is provided by the subscription registration process **1610**, downloading **1612** of additional media content may be allowed from media distribution system **18** to the personal media device **12**. Once downloaded, the new media content may be accessed **1614** with the valid user encryption key for rendering to the user. Additionally, since a valid user encryption key is present on personal media device **12**, access **1616** is provided to any pre-loaded content (via the pre-loaded SEK) present on the personal media device **12**.

[**0203**] Accordingly, the system and method allows a user to access pre-loaded content immediately after the personal media device **12** is purchased. A single encryption key (e.g., set encryption key **1306**) allows each media file included in the pre-loaded media content to be accessed and rendered. After a particular event (e.g., expiration of a trial time period, consuming a predefined playcount, etc.) access is denied to the pre-loaded content. However, by entering into a license agreement (e.g., a subscription), the user is granted access to the pre-loaded content and is allowed to download additional media content onto the personal media device and access the additional content.

[**0204**] A number of implementations have been described. Nevertheless, it will be understood that various modifications may be made. Accordingly, other implementations are within the scope of the following claims.

What is claimed is:

1. A method comprising:

allowing access to pre-loaded media content on a personal media device depending upon presence on the personal media device of a pre-loaded content encryption key associated with the pre-loaded media content.

2. The method of claim 1, wherein the pre-loaded content encryption key is stored on the personal media device prior to the personal media device being offered for sale.

3. The method of claim 1, further comprising:

denying access to the pre-loaded media content based, at least in part, upon expiration of a period of time.

4. The method of claim 1, further comprising:

denying access to the pre-loaded media content based, at least in part, upon expiration of a period of time after the media content is accessed.

5. The method of claim 1, further comprising:

denying access to pre-loaded media content based, at least in part, upon a counter that tracks accesses to the pre-loaded media content being a zero value.

6. The method of claim 1, further comprising:

allowing access to the pre-loaded media content, after the pre-loaded content encryption key has expired if a user encryption key is present on the personal media device.

7. The method of claim 6, further comprising:

allowing access to purchased media content depending upon the user encryption key on the personal media device.

8. The method of claim 6, further comprising:

allowing access to subscription media content depending upon the user encryption key on the personal media device.

9. The method of claim 1, wherein the pre-loaded content encryption key is unique to the pre-loaded media content.

10. The method of claim 6, wherein the user encryption key is unique to a subscription.

11. A method comprising:

allowing access to pre-loaded media content on a personal media device depending upon presence on the personal media device of a pre-loaded content encryption key associated with the pre-loaded media content;

decrementing a counter stored in the personal media device for each instance of media content being accessed; and

denying access to the plurality of the pre-loaded media content based, at least in part, upon if the counter reaching a predefined value.

12. The method of claim 11, further comprising:

a media distribution system, accessing the counter.

13. The method claim 11, further comprising:

allowing access to the pre-loaded media content depending upon a user encryption key being present on the personal media device.

14. A computing device configured to:

allow access to pre-loaded media content on the computing device depending upon presence on the computing device of a pre-loaded content encryption key associated with the pre-loaded media content.

15. The computing device of claim 14, wherein the pre-loaded content encryption key is stored on the computing device prior to the computing device being offered for sale.

16. The computing device of claim 14, further configured to:

deny access to the pre-loaded media content based, at least in part, upon expiration of a period of time.

17. The computing device of claim 14, further configured to:

deny access to the pre-loaded media content based, at least in part, upon expiration of a period of time after the media content is accessed.

18. The computing device of claim 14, further configured to:

deny access to the pre-loaded media content based, at least in part, upon a counter that tracks accesses to the pre-loaded media content being a zero value.

19. The computing device of claim 14, further configured to:

allow access to the pre-loaded media content, after the pre-loaded content encryption key has expired if a user encryption key is present on the computing device.

20. The computing device of claim 19, further configured to:

allow access to purchased media content depending upon the user encryption key on the computing device.

21. The computing device of claim 19, further configured for:

allowing access to subscription media content depending upon the user encryption key on the computing device.

22. The computing device of claim 14, wherein the pre-loaded content encryption key is unique to the pre-loaded media content.

23. The computing device of claim 19, wherein the user encryption key is unique to a subscription.

24. A computing device configured to:

allow access to pre-loaded media content on the computing device depending upon presence on the computing device of a pre-loaded content encryption key associated with the pre-loaded content;

decrementing a counter stored in the computing device for each instance that the media content is accessed; and

denying access to the plurality of the pre-loaded media content based, at least in part, upon if the counter reaches a predefined value.

25. The computing device of claim 24, further configured to:

a media distribution system, access the counter.

26. The computing device of claim 24, further configured to:

allow access to the pre-loaded media content depending upon a user encryption key being present on the computing device.

27. A computer program product residing on a computer readable medium having a plurality of instructions stored thereon which, when executed by a processor, cause the processor to:

allow access to pre-loaded media content on a personal media device depending upon presence on the personal media device of a pre-loaded content encryption key associated with the pre-loaded media content.

28. The computer program product of claim 27, wherein the pre-loaded content encryption key is stored on the personal media device prior to the personal media device being offered for sale.

29. The computer program product of claim 27, further comprising instructions to:

deny access to the pre-loaded media content based, at least in part, upon expiration of a period of time.

30. The computer program product of claim 27, further comprising instructions to:

deny access to the pre-loaded media content based, at least in part, upon expiration of a period of time after the media content is accessed.

31. The computer program product of claim 27, further comprising instructions to:

deny access to the pre-loaded media content based, at least in part, upon a counter that tracks accesses to the pre-loaded media content being a zero value.

32. The computer program product of claim 27, further comprising instructions to:

allow access to the pre-loaded media content, after the pre-loaded encryption key has expired if a user encryption key is present on the personal media device.

33. The computer program product of claim 32, further comprising instructions to:

allow access to purchased media content depending upon the user encryption key on the personal media device.

34. The computer program product of claim 32, further comprising instructions to:

allow access to subscription media content depending upon the user encryption key on the personal media device.

35. The computer program product of claim 27, wherein the pre-loaded content encryption key is unique to the pre-loaded media content.

36. The computer program product of claim 32, wherein the user encryption key is unique to a subscription.

37. A computer program product residing on a computer readable medium having a plurality of instructions stored thereon which, when executed by a processor, cause the processor to:

allow access to pre-loaded media content on a personal media device depending upon presence on the personal media device of a pre-loaded content encryption key associated with the pre-loaded media content;

decrement a counter stored in the personal media device for each instance that the media content is accessed; and

deny access to the pre-loaded media content based, at least in part, upon if the counter reaches a predefined value.

38. The computer program product of claim 37, further comprising instructions to:

a media distribution system, access the counter.

39. The computer program product of claim 37, further comprising instructions to:

allow access to the pre-loaded media content depending upon a user encryption key being present on the personal media device.

* * * * *